

Aram Cookson, Ryan DowlingSoka, Clinton Crumpler

Unreal Engine w 24 godziny 🔗

Nauka tworzenia gier





Tytuł oryginału: Sams Teach Yourself Unreal® Engine 4 Game Development in 24 Hours

Tłumaczenie: Joanna Zatorska Projekt okładki: Studio Gravite / Olsztyn Obarek, Pokoński, Pazdrijowski, Zaprucki

ISBN: 978-83-283-3084-9

Authorized translation from the English language edition: Sams Teach Yourself Unreal Engine 4 Game Development in 24 Hours, ISBN 0672337622; by Aram Cookson, Ryan DowlingSoka, Clinton Crumpler, and Tim Johnson; published by Pearson Education, Inc, publishing as SAMS Publishing.

Copyright © 2016 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education Inc. Polish language edition published by HELION SA. Copyright © 2017.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiejkolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Wydawnictwo HELION ul. Kościuszki 1c, 44-100 GLIWICE tel. 32 231 22 19, 32 230 98 63 e-mail: *helion@helion.pl* WWW: *http://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku! Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres *http://helion.pl/user/opinie/une24g* Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- Kup książkę
- Poleć książkę
- Oceń książkę

- Księgarnia internetowa
- Lubię to! » Nasza społeczność

Spis treści

	O autorach	. 11
	Dedykacja	. 12
	Podziękowania	. 12
	Wstęp	. 13
Godzina 1.	Wprowadzenie do Unreal Engine 4	. 17
	Instalowanie Unreal	18
	Tworzenie pierwszego projektu	20
	Interfejs użytkownika	22
	Tryby widoku i wizualizatory	29
	Testowanie określonego poziomu gry	32
	Podsumowanie	32
	Pytania i odpowiedzi	33
	Pytania sprawdzające	33
	Ćwiczenia	34
Godzina 2.	Platforma Gameplay	. 35
	Dostępne zasoby	35
	Referencje zasobów i okno Reference Viewer	43
	Platforma Gameplay	44
	Podsumowanie	48
	Pytania i odpowiedzi	48
	Pytania sprawdzające	49
	Ćwiczenie	49
Godzina 3.	Współrzędne, transformacje, jednostki i organizacja	. 51
	Współrzędne kartezjańskie	51
	Wykorzystanie transformacji	52
	Określanie jednostek i dokonywanie pomiarów	55
	Organizowanie sceny	57

	Podsumowanie	63
	Pytania i odpowiedzi	63
	Pytania sprawdzające	64
	Ćwiczenie	64
Godzina 4.	Korzystanie z aktorów typu Static Mesh	67
	Zasoby typu Static Mesh	67
	Edytor siatek statycznych	
	Wyświetlanie układów UV	71
	Siatki kolizji	73
	Aktorzy typu Static Mesh	
	Podsumowanie	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 5.	Oświetlenie i renderowanie	87
	Terminologia związana z oświetleniem	
	Typy oświetlenia	
	Właściwości oświetlenia	
	Budowanie oświetlenia	94
	Podsumowanie	97
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 6.	Używanie materiałów	101
	Koncepcja materiałów	
	Renderowanie oparte na prawach fizyki	
	Typy materiałów wejściowych	
	Tworzenie tekstur	105
	Tworzenie materiału	
	Podsumowanie	113
	Pytania i odpowiedzi	
	Pytania sprawdzające	116
	Ćwiczenie	116
Godzina 7.	Używanie elementów systemu dźwięków	119
	Podstawowe zagadnienia związane z dźwiękami	119
	Używanie aktorów typu Sound	122

	Kontrolowanie dźwięków za pomocą obszarów w przestrzeni	127
	Podsumowanie	128
	Pytania i odpowiedzi	129
	Pytania sprawdzające	130
	Ćwiczenie	130
Godzina 8.	Tworzenie krajobrazów i roślinności	133
	Tworzenie krajobrazów	133
	Rzeźbienie kształtów i przestrzeni	137
	Tworzenie roślinności	142
	Podsumowanie	145
	Pytania i odpowiedzi	145
	Pytania sprawdzające	
	Ćwiczenie	146
Godzina 9.	Budowanie świata	147
	Budowanie światów	148
	Proces budowania świata	149
	Podsumowanie	162
	Pytania i odpowiedzi	
	Pytania sprawdzające	165
	Ćwiczenie	165
Godzina 10.	Tworzenie efektów za pomocą systemów cząstek	167
	Cząstki i typy danych	167
	Korzystanie z narzędzia Cascade	168
	Korzystanie z popularnych modułów	174
	Konfigurowanie materiałów dla cząstek	178
	Wyzwalanie systemów cząsteczek	182
	Podsumowanie	182
	Pytania i odpowiedzi	
	Pytania sprawdzające	183
	Ćwiczenie	184
Godzina 11.	Używanie aktorów siatki szkieletowej	185
	Definiowanie siatek szkieletowych	185
	Importowanie siatek szkieletowych	188
	Używanie edytora Persona	193

	Użycie aktorów siatki szkieletowej	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 12.	Tworzenie klipów filmowych	205
	Aktorzy typu Matinee	
	Edytor Matinee	
	Curve Editor	
	Korzystanie z innych ścieżek	
	Praca z kamerami w klipie Matinee	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 13.	Przestrzeganie praw fizyki w grze	225
	Korzystanie z praw fizyki w UE4	
	Symulowanie właściwości fizycznych	
	Używanie materiałów fizycznych	231
	Korzystanie z ograniczeń	
	Korzystanie z aktorów typu Force	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 14.	Wprowadzenie do wizualnego systemu	
	skryptowego Blueprint	245
	Podstawy tworzenia skryptów metodą wizualną	245
	Używanie edytora Blueprint	
	Podstawowe koncepcje związane ze skryptami	252
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	

Godzina 15.	Skrypty Blueprint poziomu	
	Ustawienia kolizji aktora	
	Przydzielanie aktorów do zdarzeń	
	Przydzielanie aktorów do zmiennych referencyjnych	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 16.	Korzystanie z klas Blueprint	285
	Używanie klas Blueprint	
	Interfejs edytora Blueprint	
	Praca z komponentami	
	Korzystanie z osi czasu	
	Tworzenie skryptu pulsującego światła	298
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	305
	Ćwiczenie	305
Godzina 17.	Zmienne edytowalne i skrypt konstrukcyjny	307
	Konfiguracja	
	Tworzenie zmiennych edytowalnych	
	Używanie skryptu konstrukcyjnego	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	
	Ćwiczenie	
Godzina 18.	Obsługa zdarzeń klawiatury i rozmieszczanie aktorów .	319
	Dlaczego rozmieszczanie obiektów jest ważne	
	Tworzenie klasy Blueprint rozmieszczanych aktorów	
	Konfigurowanie obiektu Blueprint, który bedzie rozmieszczać aktorów	
	Rozmieszczanie aktora określonej klasy	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzajace	
	Ćwiczenie	

Godzina 19. ⁻	Гworzenie interakcji	333
	Tryby gry projektu	333
	Znajomość możliwości postaci	334
	Używanie klas Blueprint	336
	Znaczniki aktora i komponentu	342
	Podsumowanie	343
	Pytania i odpowiedzi	343
	Pytania sprawdzające	343
	Ćwiczenie	344
Godzina 20.	Tworzenie strzelanki platformowej:	
	systemy wejścia i postacie	345
	Identyfikowanie wymagań na podstawie podsumowania projektu	346
	Tworzenie projektu gry	346
	Tworzenie niestandardowego trybu gry	348
	Tworzenie własnego kontrolera postaci i gracza	351
	Kontrolowanie ruchu obiektu Pawn	353
	Konfiguracja kamery o stałej pozycji	360
	Podsumowanie	362
	Pytania i odpowiedzi	363
	Pytania sprawdzające	364
	Ćwiczenie	364
Godzina 21.	Tworzenie strzelanki platformowej:	
	przeszkody i obiekty do zebrania	367
	Tworzenie bazowej klasy przeszkody	368
	Wprawianie przeszkody w ruch	370
	Uszkadzanie obiektu typu Pawn	374
	Ponowne uruchomienie gry po śmierci gracza	378
	Tworzenie uzdrawiających obiektów	380
	Tworzenie obiektu rozmieszczającego aktorów	386
	Usuwanie starych przeszkód	391
	Podsumowanie	391
	Pytania i odpowiedzi	391
	Pytania sprawdzające	392
	Ćwiczenie	393

Godzina 22.	Korzystanie z UMG	395
	Tworzenie widżetu Blueprint	395
	Nawigacja w interfejsie UMG	396
	Tworzenie menu Start	401
	Przykładowy system menu	411
	Podsumowanie	
	Pytania i odpowiedzi	412
	Pytania sprawdzające	413
	Ćwiczenie	413
Godzina 23.	Tworzenie pakietu wykonywalnego	415
	Opracowywanie zawartości	415
	Pakowanie projektu dla systemu Windows	416
	Zasoby potrzebne do utworzenia pakietu przeznaczonego dla systemów Android i iOS	
	Dostep do zaawansowanych ustawień pakietów	
	Podsumowanie	
	Pytania i odpowiedzi	
	Pytania sprawdzające	423
	Ćwiczenie	
Godzina 24.	Projekty dla urządzeń mobilnych	425
	Tworzenie gier dla urządzeń mobilnych	426
	Korzystanie z interfejsu dotykowego	436
	Użycie danych o ruchu urządzenia	441
	Podsumowanie	
	Pytania i odpowiedzi	444
	Pytania sprawdzające	445
	Ćwiczenie	446
	Skorowidz	

10 Spis treści

Godzina 1.

Wprowadzenie do Unreal Engine 4

W trakcie tej godziny nauczysz się:

- Instalować aplikację Epic Games Launcher
- Instalować Unreal Engine
- Tworzyć nowy projekt
- Korzystać z interfejsu edytora Unreal Engine

Witamy w Unreal Engine! Unreal Engine 4 (UE4) jest silnikiem gier i edytorem opracowanym przez Epic Games. Służy do tworzenia różnorodnych gier i aplikacji, począwszy od wysokiej jakości gier na konsole wideo, po niezależne aplikacje na urządzenia mobilne. Unreal Engine jest zgodny z systemem operacyjnym Windows i Mac. Umożliwia tworzenie aplikacji działających na platformach Windows, Mac, PlayStation 4, Xbox One, iOS, Android, HTML5 i Linux. W najprostszej postaci Unreal Engine 4 jest kolekcją edytorów wykorzystywanych przez różne branże do produkcji gier lub aplikacji.

W trakcie tej godziny dowiesz się, jak pobrać i zainstalować Unreal Engine oraz stworzyć pierwszy projekt. Zapoznasz się też z interfejsem edytora. Zaczniesz od utworzenia konta użytkownika, a następnie pobierzesz i zainstalujesz aplikację Epic Games Launcher. Posłuży ona do pobrania UE4. Gdy wszystko będzie gotowe, utworzysz pierwszy projekt, nauczysz się korzystać z interfejsu edytora i dowiesz się, jak poruszać się po poziomie gry. Przetestujesz też domyślną mapę.

Uwaga

Unreal Engine jest bezpłatny!

Tak jest, UE4 jest całkowicie bezpłatny! Wszystko jest dostępne za darmo! Dlaczego Epic Games udostępnia to wszystko za darmo? Nigdy nie wiadomo, gdzie narodzi się następna fantastyczna gra lub aplikacja. Jakie korzyści czerpie z tego Epic? Gdy udostępnisz grę i zaczniesz na niej zarabiać, będziesz musiał zapłacić firmie Epic 5% przychodów. Oczywiście, szczegóły można znaleźć na stronie Epic. Epic udostępnia też sklep, w którym można kupić i pobrać zasoby dla swoich projektów. Nie musisz z niego korzystać, ponieważ wszystko możesz oczywiście utworzyć samodzielnie, często jednak warto pójść przetartym szlakiem, aby przyspieszyć produkcję.

Instalowanie Unreal

Instalowanie Unreal Engine jest prostym procesem, składającym się z następujących trzech etapów:

- 1. Tworzenie nowego konta użytkownika.
- 2. Pobieranie i instalowanie aplikacji Epic Games Launcher.
- 3. Pobieranie Unreal Engine.

Pobieranie i instalowanie programu Launcher

Program Launcher ułatwia zarządzanie wersjami Unreal Engine, które są zainstalowane na komputerze. Za jego pomocą można zarządzać projektami, uzyskać dostęp do darmowych projektów przykładowych i przejść do sklepu, w którym można zakupić zasoby do wykorzystania we własnych projektach. Program ten wyświetla też wiadomości ze społeczności skupionej wokół UE i zawiera łącza do zasobów szkoleniowych oraz dokumentacji dostępnych online.

Uwaga

Uwaga

Wymagania dotyczące systemów operacyjnych i sprzętu

Aby efektywnie wykorzystać Unreal Engine, potrzebny jest komputer działający pod kontrolą systemu Windows lub Mac OS i spełniający następujące kryteria:

- System operacyjny: 64-bitowy Windows 7 lub 8 albo Mac OS X 10.9.2
- Procesor: Quad-core Intel lub AMD, 2.5 GHz lub szybszy
- Karta graficzna: NVIDIA GeForce 470 GTX lub AMD Radeon w wersji 6870 lub wyższej
- Pamięć: 8 GB RAM

Wykonaj następujące czynności, aby pobrać i zainstalować Epic Games Launcher:

- 1. Otwórz serwis Unreal Engine (www.unrealengine.com), widoczny na rysunku 1.1.
- 2. Kliknij przycisk Get Unreal.
- 3. Na stronie, która się pojawi, utwórz nowe konto użytkownika.
- 4. Wybierz wersję dla systemu Windows lub Mac OS, w zależności od swojego systemu operacyjnego, i pobierz plik instalacyjny (o rozszerzeniu *.msi* w przypadku systemu Windows lub *.dmg* w przypadku systemu Mac).
- **5.** Uruchom instalator, wybierz lokalizację instalacji i wykonaj instrukcje wyświetlane na ekranie.



RYSUNEK 1.1. Serwis internetowy Unreal Engine

Wskazówka

Miejsce na dysku twardym

Silniki gier zajmują bardzo dużo przestrzeni na dysku twardym. Po pobraniu nowa wersja UE4 zostanie automatycznie zainstalowana w tym samym miejscu, w którym znajduje się aplikacja Launcher. Dlatego podczas instalowania aplikacji Launcher wybierz folder, w którym jest do dyspozycji co najmniej 20 GB wolnego miejsca. Chociaż wstępna instalacja nie wymaga tak dużej przestrzeni na dysku, po pobraniu przykładowej zawartości lub zakupie zasobów w sklepie wymagania dotyczące wolnego miejsca będą wzrastać. Na szczęście, własne projekty można zapisać w dowolnym miejscu.

Pobieranie i instalowanie Unreal Engine

Po zainstalowaniu aplikacji Launcher można za jej pomocą pobrać i zainstalować UE4. UE4 zajmuje znacznie więcej miejsca niż Launcher, a jego pobieranie potrwa kilka minut. Epic zawsze ulepsza swoje oprogramowanie i często publikuje nowe wersje UE4. Najlepiej zacząć naukę od zainstalowania najnowszej wersji oficjalnej.

Ostrzeżenie

Ostrzeżenie

Wersje poglądowe

Epic często publikuje wersje poglądowe kolejnych wersji programu. Umożliwia w ten sposób przetestowanie wczesnych wersji pod kątem ewentualnych błędów. Jeśli dopiero zaczynasz naukę UE4, najlepiej pobierz najnowszą wersję, która nie jest wersją poglądową. Zawsze możesz uaktualnić projekty do kolejnej wersji, gdy tylko będzie dostępna. Pobieranie i instalowanie UE4 za pomocą aplikacji Epic Games Launcher odbywa się następująco:

- 1. Otwórz aplikację Launcher i wybierz kategorię Library (Biblioteka) (patrz rysunek 1.2).
- 2. Przejdź do sekcji zawierającej wersje silnika Unreal Engine i kliknij przycisk *Add Version (Dodaj wersje)*, aby dodać element nowej wersji.
- **3.** Przejdź do elementu dotyczącego nowej wersji, kliknij znajdującą się tam listę i wybierz pożądaną wersję.
- **4.** Kliknij *Install (Zainstaluj)*, aby Launcher pobrał i zainstalował Unreal Engine. UE4 jest ogromny, dlatego jego pobieranie zajmie nieco czasu.



RYSUNEK 1.2. Kategoria Library (Biblioteka) w aplikacji Launcher

Tworzenie pierwszego projektu

Po pobraniu UE4 nadszedł czas na utworzenie pierwszego projektu. Podczas pierwszego uruchomienia UE4 na ekranie pojawi się okno *Project Browser* (przeglądarka projektu). W oknie tym znajduje się zakładka *Projects* (projekty), gdzie widoczne są wszystkie projekty, nad którymi pracujesz. Dostępna jest też zakładka *New Project* (nowy projekt), która umożliwia tworzenie nowych projektów na podstawie istniejących popularnych szablonów trybów gier.

IDŹ DO ► Godzina 2., "Platforma Gameplay", aby zapoznać się ze szczegółowymi informacjami na temat anatomii projektu.

Okno Project Browser

Podczas tworzenia nowego projektu w oknie *Project Browser* należy podjąć kilka decyzji. Po pierwsze, musisz wybrać, czy chcesz utworzyć projekt oparty na systemie Blueprint lub na języku C++. W tej książce będziemy wykorzystywać tylko system skryptowy Blueprint. Musisz też wybrać docelową platformę. Do wyboru masz opcję *Desktop/Console* (komputer stacjonarny/konsola) lub *Mobile/Tablet* (urządzenie mobilne/tablet). Możesz też wybrać opcje związane z grafiką, czyli *Maximum Quality* (maksymalna jakość) lub *Scalable 3D or 2D* (skalowalna grafika 3D lub 2D). Te opcje zmieniają domyślne ustawienia projektu dotyczące tworzenia zawartości gry. Na koniec musisz podjąć decyzję, czy chcesz skorzystać z domyślnej zawartości graficznej. Jeśli zaznaczysz opcję *With Starter Content* (z zawartością początkową), będziesz mieć do dyspozycji gotowe zasoby.

Uwaga

Projekty oparte na systemie Blueprint lub na C++

Blueprint jest wizualnym środowiskiem skryptowym, wykorzystywanym do tworzenia funkcjonalności projektu gry. Projekty oparte na C++ umożliwiają tworzenie funkcjonalności poprzez tradycyjny kod. Projekt oparty na C++ wymaga zainstalowania kompilatora, na przykład Visual Studio 2013. Jeśli wcześniej nie pisałeś kodu lub nie używałeś jeszcze UE4, warto najpierw zapoznać się z edytorem i sposobem jego obsługi. Później możesz przejść do tworzenia projektów na podstawie C++.

Ostrzeżenie

Ostrzeżenie

Rozmiar projektu

Nowe projekty początkowo nie są zbyt duże, lecz ich rozmiary szybko rosną. Zależy to od ilości i jakości zawartości, na przykład od szczegółowości modeli i rozmiarów tekstur. UE4 generuje też pliki podczas automatycznego zapisywania i tworzenia kopii zapasowych; chociaż zajmują one sporo miejsca, umożliwiają odzyskanie pracy po wystąpieniu awarii.

🔻 SPRÓBUJ SAM

Tworzenie projektu

Projekty można tworzyć w dowolnym folderze, w którym dostępna jest wystarczająca ilość wolnego miejsca. Wybierz folder, w którym dostępne jest co najmniej 2 GB wolnego miejsca i który łatwo znajdziesz. Na rysunku 1.3 widoczne są podstawowe ustawienia pierwszego projektu, a poniżej znajdują się kroki wymagane do jego utworzenia:

- 1. W aplikacji Launcher kliknij przycisk Launch (Uruchom).
- 2. Wybierz zakładkę New Project.
- 3. Wybierz zakładkę Blueprint.
- 4. Zaznacz szablon First Person (pierwsza osoba).
- 5. Wybierz docelową platformę Desktop/Console.
- 6. Wybierz jakość docelowej grafiki Scalable 3D or 2D.
- 7. Wybierz opcję With Starter Content.

- 8. Wybierz folder, w którym dostępne jest kilka gigabajtów wolnego miejsca.
- 9. Nazwij projekt.



RYSUNEK 1.3. Ustawienia służące do utworzenia pierwszego projektu

10. Kliknij przycisk Create Project (utwórz projekt).

Uwaga

Modyfikowanie ustawień projektu

Możesz zmienić niektóre ustawienia istniejącego projektu, takie jak docelowa platforma i jakość grafiki. W tym celu otwórz okno *Project Settings* (ustawienia projektu), wybierając opcję o tej nazwie z menu *Edit* (edycja).

Interfejs użytkownika

Po zainstalowaniu UE4 i utworzeniu pierwszego projektu możesz się zabrać do dzieła. Jak już wcześniej wspomniano, UE4 składa się ze zbioru edytorów oraz narzędzi, wykorzystywanych podczas tworzenia różnych aspektów gry. Na początku skoncentrujemy się na nauce podstawowych elementów głównego interfejsu i sposobów poruszania się po poziomie gry. Główny interfejs, zwany edytorem poziomu (*Level Editor*), służy przede wszystkim do tworzenia świata i poziomów gry, a także do rozmieszczania zasobów.

Interfejs edytora składa się z kilku kluczowych paneli, które powinieneś poznać. Jest to pasek menu, panel *Modes* (tryby), panel *World Outliner* (zarys świata), panel *Details* (szczegóły), panel *Content Browser* (przeglądarka zawartości), pasek narzędzi oraz panel *Viewport* (okno podglądu) (patrz rysunek 1.4). W kolejnych punktach znajduje się opis poszczególnych paneli.



RYSUNEK 1.4. Domyślny interfejs edytora UE4

Uwaga

Układ elementów interfejsu

Układ elementów interfejsu edytora możesz dostosować do swoich potrzeb, z łatwością zmieniając ułożenie paneli i okien. W tej książce wszystkie elementy interfejsu znajdują się w domyślnym położeniu, lecz gdy zdobędziesz nieco więcej doświadczenia, prawdopodobnie zechcesz zmienić położenie elementów, aby ułatwić sobie pracę.

Pasek menu

Na pasku menu, podobnie jak w innych nowoczesnych aplikacjach, znajduje się menu *File* (plik), *Edit* (edycja), *Window* (okno) i *Help* (pomoc). Menu *File* zawiera operacje służące do wczytywania i zapisywania projektów i poziomów gier. Menu *Edit* zawiera standardowe operacje kopiowania i wklejania, a także preferencje edytora i ustawienia projektu. Menu *Window* służy do otwierania paneli *Viewport* i innych. Jeśli zamkniesz jakieś okno lub panel, możesz je ponownie otworzyć za pomocą menu *Window*. Menu *Help* zawiera łącza do zewnętrznych zasobów, takich jak dokumentacja i samouczki online.

Panel Modes

Panel *Modes* wyświetla różne tryby edycji dostępne w edytorze poziomu (patrz rysunek 1.5). Umożliwia włączenie wyspecjalizowanych interfejsów, w których możesz tworzyć niektóre rodzaje aktorów i brył.



RYSUNEK 1.5. Panel Modes

Wskazówka

Wskazówka

Co to jest aktor?

Najważniejsze w nauce obsługi dowolnego oprogramowania jest poznanie jego interfejsu, sposobów pracy i terminologii. Z silnikiem Unreal związana jest ogromna ilość terminologii, z którą zetkniesz się również w tej książce. Termin *aktor* odnosi się do zasobu, który został umieszczony w poziomie gry. Na przykład, model 3D w panelu *Content Browser* jest *zasobem siatki statycznej (Static Mesh)*. Lecz po umieszczeniu instancji zasobu siatki statycznej w poziomie instancję tę nazywany *aktorem siatki statycznej*.

Panel *Modes* zawiera różne tryby narzędzi edytora poziomów, które zmieniają sposób jego działania. Za pomocą tego panelu możesz wykonać szczególne zadanie, na przykład umieścić w świecie nowy zasób, wyrzeźbić krajobraz, utworzyć pędzle geometrii i brył, utworzyć roślinność lub namalować siatki. W tabeli 1.1 opisana jest lista trybów edytora oraz efekt uzyskany po ich wybraniu.

Akcja	Efekt
Tryb <i>Place</i> (umieszczanie)	Służy do umieszczania aktorów na scenie.
Tryb <i>Paint</i> (malowanie)	Służy do malowania kolorów wierzchołków aktorów siatki statycznej.
Tryb <i>Landscape</i> (krajobraz)	Służy do edycji aktorów terenu krajobrazu.
Tryb <i>Foliage</i> (roślinność)	Służy do malowania instancji aktorów roślinności w poziomie gry.
Tryb <i>Geometry Editing</i> (edytowanie geometrii)	Służy do edycji powierzchni brył aktorów pędzli BSP.

TABELA 1.1. Tryby edytora

Panel World Outliner

Panel *World Outliner* wyświetla drzewo hierarchii wszystkich aktorów znajdujących się w bieżącym poziomie gry (patrz rysunek 1.6). Aktora można zaznaczyć, klikając jego nazwę w panelu *World Outliner*. W panelu *Details* zostaną wówczas wyświetlone jego właściwości. Jeśli dwukrotnie klikniesz nazwę zasobu, zostanie on zaznaczony i wyśrodkowany w panelu *Viewpoint*.



RYSUNEK 1.6. Panel World Outliner

Panel Details

Panel *Details* jest jednym z najczęściej wykorzystywanych elementów interfejsu UE4. Panel ten znajduje się niemal w każdym z edytorów. W panelu *Details* wyświetlone są wszystkie edytowalne właściwości aktorów zaznaczonych w panelu *Viewport.* Wyświetlone właściwości zależą od typu zaznaczonego aktora, lecz istnieje kilka właściwości, które dotyczą większości aktorów (patrz rysunek 1.7). Do typowych właściwości należą nazwa aktora, pola edycji transformacji przesunięcia, obrotu i skalowania aktorów, a także właściwości renderowania.

Uwaga

Zaznaczanie aktorów

Aby zaznaczyć aktora, kliknij go w panelu *Viewport* lub w panelu *World Outliner*. Aktor zostanie podświetlony, a w panelu *Details* zostaną wyświetlone jego podstawowe właściwości. Jednocześnie można zaznaczyć kilku aktorów — w tym celu należy wykonać poniższe czynności:

- W panelu Viewport i World Outliner przytrzymaj klawisz Ctrl lub Shift i dodaj lub usuń z zaznaczenia kolejnych aktorów.
- Tylko w panelu Viewport możesz nacisnąć klawisze Ctrl+Alt i kliknąć, a następnie przeciągnąć ramkę zaznaczenia wokół kilku aktorów.

🗓 Details 🛛 🛛	left World Settings	
CargeCube	0	6
+ Add Component -		📽 Blueprint/Add Script
🙀 LargeCube(Instance)		
n StaticMeshComponent	(Inherited)	
Search		∽ه 🏢 🛛
▲ Transform		
Location 🛩	X 530.353638 c Y -1169	9.604492 Z 257.000519 c つ
Rotation 🔫	X 0.0* 🖍 Y 0.0*	S Z 0.0° S
Scale 🔫	X 1.0 Y 1.0	Z 1.0
Mobility	💿 Static 🖨 Mova	ible a
▲ Static Mesh		
Static Mesh	FirstPerson ← ♪	Cube_Rounded 👻 5
▲ Materials		
Element 0	MI_Templar ← ♀ ゥ Textures ←	te_BaseGray_02
▲ Physics		
Simulate Physics	2 °	
Mass in Kg	2211.21582 Override	
Linear Damping	0.01	
Angular Damping	0.0	
Enable Gravity		
Constraints		

RYSUNEK 1.7. Panel Details głównego edytora

Panel Content Browser

Panel *Content Browser* jest podstawowym narzędziem służącym do zarządzania zasobami w projekcie (patrz rysunek 1.8). Panel ten umożliwia wykonanie ogólnych zadań związanych z zawartością gry, takich jak tworzenie, przeglądanie, modyfikowanie, importowanie i organizowanie. Panel *Content Browser* umożliwia też zarządzanie folderami i wykonywanie podstawowych operacji na zasobach, takich jak przeglądanie referencji, przenoszenie, kopiowanie i zmiana nazwy. Panel ten zawiera też pasek wyszukiwania i flagi filtrowania, które umożliwiają szybkie znajdowanie zasobów.

Panel *Content Browser* można postrzegać jako nieskończenie pojemne pudło na zasoby. Gdy potrzebujesz coś umieścić w grze, możesz pobrać *instancję* (czyli kopię) zasobu z tego pudełka i umieścić ją w poziomie. Instancja po umieszczeniu w poziomie staje się aktorem, który jest dokładną kopią oryginalnego zasobu znajdującego się w panelu *Content Browser*. Umieszczonego w grze aktora można zmodyfikować za pomocą panelu *Details*. Z lewej strony panelu *Content Browser* znajduje się panel *Source* (źródła), w którym wyświetlona



RYSUNEK 1.8. Panel Content Browser. Z lewej strony widoczne są foldery źródłowe, a z prawej — obszar zarządzania zasobami

jest hierarchia folderów z zawartością. Panel *Source* można rozwinąć lub zwinąć, klikając ikonę w lewym górnym rogu, poniżej przycisku *Add New* (dodaj nowy). Prawa strona panelu *Content Browser* nazywana jest obszarem zarządzania zasobami. Wyświetlone są tu zasoby znajdujące się w folderze zaznaczonym w panelu *Source*.

Wskazówka

Wskazówka

Organizacja folderów

Złożoność projektów może szybko wzrosnąć, dlatego dobra organizacja plików jest kluczowa w utrzymaniu efektywnego środowiska pracy. Zaleca się umieszczanie zasobów różnego typu w osobnych folderach. Foldery można zagnieżdżać, aby uzy-skać maksymalny porządek i wydajność pracy.

Panel Viewport

Viewport, czyli okno podglądu, jest swego rodzaju oknem na świat tworzonej przez Ciebie gry. Panel *Viewport* służy do poruszania się po bieżącym poziomie. W panelu tym mamy do dyspozycji wiele różnych trybów, układów i ustawień, które ułatwiają tworzenie i edycję poziomów, a także zarządzanie nimi (patrz rysunek 1.9).



RYSUNEK 1.9. Panel Viewport

Układ panelu Viewport

28

Domyślnie panel *Viewport* zawiera jedno okno z widokiem *Perspective* (perspektywa). Możesz jednak z łatwością zwiększyć liczbę okien do dwóch, trzech lub czterech, klikając listę w lewym górnym rogu panelu. Wybierz z listy opcję *Layouts* (układy), a następnie pożądany format (patrz rysunek 1.10). Tryb widoku możesz wybrać w każdym oknie znajdującym się w panelu *Viewport*.



RYSUNEK 1.10. Opcje układu panelu Viewport

Typy widoków

W panelu *Viewport* dostępne są dwa podstawowe typy widoków — *Perspective* i *Orthographic* (ortograficzny) (patrz rysunek 1.11). Widok typu *Perspective* przedstawia trójwymiarowy świat z punktami zbiegu perspektywy, zaś widok typu *Orthographic* przedstawia dwuwymiarowy schemat świata. Prawdopodobnie najczęściej będziesz korzystać z widoku *Perspective*, lecz widoki typu *Orthographic* doskonale nadają się do precyzyjnego rozmieszczania aktorów na scenie.



RYSUNEK 1.11. Ustawienia typu widoku w panelu Viewport

Tryby widoku i wizualizatory

Tryby widoku (patrz rysunek 1.12) zmieniają wygląd świata w oknie podglądu i są niezależne od typu widoku. Mogą być źródłem ważnych informacji o stanie poziomu. W tabeli 1.2 opisano popularne tryby widoku.







Wizualizacja

Do dyspozycji mamy ponad 13 różnych trybów widoku oraz innych narzędzi wizualizacji. Za ich pomocą można uzyskać dodatkowe informacje dotyczące poziomu oraz zdebugować grę i rozwiązać potencjalne błędy.

Tryb	Efekt
<i>Lit</i> (oświetlony)	Przedstawia ostateczny wygląd sceny po zastosowaniu materiałów i oświetlenia.
<i>Unlit</i> (nieoświetlony)	Usuwa całe oświetlenie ze sceny, pokazując kolor bazowy przydzielonych materiałów.
<i>Wireframe</i> (szkielet)	Pokazuje krawędzie wielokątów, z których zbudowani są aktorzy znajdujący się na scenie.
<i>Detail Lighting</i> (oświetlenie szczegółów)	Wyświetla neutralny materiał w całej scenie, wykorzystując mapy normalnych z przydzielonych materiałów.
<i>Lighting Only</i> (samo oświetlenie)	Wyświetla neutralny materiał, na który wpływ ma tylko oświetlenie, z pominięciem danych zawartych w mapach normalnych.

TABELA 1.2. Popularne tryby widoku w panelu Viewport

Flagi na liście Show

Flagi na liście *Show* (pokaż) umożliwiają wyświetlanie odpowiednich informacji bezpośrednio w panelu *Viewport*. Możesz na przykład włączyć wyświetlanie siatki wykrywania kolizji aktora lub jego obwiedni.

Nawigacja po scenie w oknie podglądu perspektywy

Po wstępnym zapoznaniu się z kluczowymi obszarami głównego interfejsu musisz się nauczyć poruszania po poziomie w panelu *Viewport*. W tabelach 1.3 i 1.4 opisano najpopularniejsze sposoby nawigacji w poziomie wyświetlonym w panelu *Viewport*.

Sposób	Akcja	
Widok typu Perspective		
Kliknij i przeciągnij	Przesuwa kamerę panelu <i>Viewport</i> do przodu i do tyłu oraz obraca ją w lewo i w prawo.	
Kliknij prawym przyciskiem myszy i przeciągnij	Obraca kamerę okna podglądu w miejscu, bez jej przesuwania do przodu lub do tyłu.	
Kliknij, po czym kliknij prawym przyciskiem myszy i przeciągnij	Przesuwa kamerę okna podglądu w górę i w dół.	
Widoki typu <i>Orthographic</i> (<i>Top</i> (z góry), <i>Fro</i>	<i>nt</i> (z przodu), <i>Side</i> (z boku))	
Kliknij i przeciągnij	Tworzy ramkę zaznaczenia.	
Kliknij prawym przyciskiem myszy i przeciągnij	Przesuwa podgląd w lewo i w prawo.	
Kliknij, po czym kliknij prawym przyciskiem myszy i przeciągnij	Powiększa i pomniejsza podgląd.	

 TABELA 1.3. Sposoby poruszania się w panelu Viewport

-

Uwaga

Nawigacja w poziomie gry

W przeciwieństwie do aplikacji modelowania 3D, które koncentrują się na jednym, tworzonym obiekcie i obracają się wokół niego, sposoby poruszania się w oknie podglądu Unreal Engine są przystosowane do tworzenia ogromnych poziomów gry. W tym przypadku możliwość szybkiego poruszania się po wielkich obszarach jest kluczowa.

TABELA 1.4. Orbit	owanie, najeżdżar	nie kamerą i śledzeni	e w oknie podglądu
-------------------	-------------------	-----------------------	--------------------

Sposób	Akcja
Naciśnij klawisz F	Naciśnięcie klawisza <i>F</i> spowoduje skierowanie kamery na aktora zaznaczonego w panelu <i>Viewport</i> .
Naciśnij klawisz <i>Alt</i> oraz lewy przycisk myszy i przeciągnij	Obraca okno podglądu wokół środka obiektu.
Naciśnij klawisz <i>Alt</i> oraz prawy przycisk myszy i przeciągnij	Nakierowuje (zbliża) lub oddala kamerę od środka obiektu.
Naciśnij klawisz <i>Alt</i> oraz środkowy przycisk myszy i przeciągnij	Śledzi kamerą obraz w lewo, w prawo, w górę i w dół, zgodnie z ruchami myszy.

Wskazówka

Nawigacja w sposób stosowany w grze

Jeśli w widoku *Perspective* przytrzymasz prawy przycisk myszy, możesz poruszać się w poziomie, naciskając klawisze *W*, *A*, *S* i *D*. W podobny sposób porusza się strzelec w grze w perspektywie pierwszej osoby.

Pasek narzędzi edytora poziomu

Pasek narzędzi edytora poziomu zapewnia szybki dostęp do popularnych narzędzi i operacji, takich jak zapisywanie bieżącego poziomu, tworzenie wstępnego oświetlenia aktorów statycznych, zmiana właściwości wyświetlania edytora i testowanie gry w bieżącym poziomie. Pasek narzędzi edytora poziomu jest przedstawiony na rysunku 1.13.



RYSUNEK 1.13. Pasek narzędzi edytora poziomu

Testowanie określonego poziomu gry

Gdy utworzysz nowy projekt, masz już do dyspozycji domyślny poziom. Jest on jedną z pierwszych rzeczy widocznych w edytorze po otwarciu projektu. Testowanie gry wymaga skorzystania z takiego samego systemu wejścia, jakim będą dysponować gracze. Istnieje kilka trybów gry testowej w poziomie (PIE — ang. *Play in Editor*) (patrz rysunek 1.14). Na tym etapie nauki wypróbuj podstawowe tryby gry testowej w edytorze, czyli *Selected Viewport* (zaznaczone okno podglądu) i *New Editor Window* (nowe okno edytora). Aby przetestować poziom, kliknij ikonę *Play* (odtwórz). Jeśli chcesz wybrać konkretny tryb, kliknij strzałkę w dół, znajdującą się obok wspomnianej ikony.



RYSUNEK 1.14. Tryby gry testowej

Wskazówka

Wskazówka

Testowanie poziomu gry

Ostatni wybrany tryb gry testowej stanie się automatycznie domyślnym trybem na pasku narzędzi edytora poziomów. Jeśli nie chcesz używać domyślnego trybu gry testowej, kliknij strzałkę obok ikony *Play* i wybierz inny tryb.

Podsumowanie

W trakcie tej godziny pobrałeś i zainstalowałeś Unreal Engine, zapoznałeś się z kluczowymi elementami głównego interfejsu, utworzyłeś pierwszy projekt, nauczyłeś się poruszać w poziomie gry w panelu *Viewport* i dowiedziałeś się, jak testować poziom gry. Im lepiej opanujesz te zadania, tym lepiej dla Ciebie.

Pytania i odpowiedzi

P. Do czego służy aplikacja Epic Games Launcher?

O. Epic Games Launcher umożliwia zarządzanie projektami i daje dostęp do sklepu, w którym można kupić zasoby. Aplikacja ta umożliwia też aktualizowanie UE4.

P. Gdzie jest instalowany UE4?

O. UE4 jest instalowany w tym samym miejscu, co Launcher.

P. Gdzie powinienem zapisywać projekty?

O. Projekty należy zapisywać na dysku twardym, na którym dostępna jest wystarczająca ilość wolnego miejsca i który jest łatwo dostępny.

Pytania sprawdzające

Po zapoznaniu się z materiałem tego rozdziału sprawdź, czy potrafisz odpowiedzieć na poniższe pytania.

Quiz

- 1. Prawda czy fałsz? Panel Modes umożliwia włączanie różnych trybów edycji.
- **2.** Jeśli w panelu *Viewport* chcesz skierować podgląd na zaznaczonego aktora, z jakiego klawisza lub kombinacji klawiszy skorzystasz?
- 3. Prawda czy fałsz? Dowolny zasób umieszczony w poziomie jest nazywany aktorem.
- 4. Do zarządzania nowymi projektami i ich tworzenia używamy _____
- 5. Prawda czy fałsz? Układ głównego interfejsu można całkowicie zmienić.
- 6. Prawda czy fałsz? Skrót PIE pochodzi od słów Play in Editor.

Odpowiedzi

- 1. Prawda. Panel *Modes* umożliwia włączenie trybu umieszczania aktorów, tworzenia terenu i tworzenia roślinności.
- **2.** Naciśnięcie klawisza *F* spowoduje skierowanie kamery na aktora zaznaczonego w panelu *Viewport*.
- **3.** Prawda. Zasób dowolnego typu umieszczony w poziomie staje się instancją oryginalnego zasobu i jest nazywany aktorem.
- **4.** *Project Browser*. Jednocześnie możesz pracować nad wieloma projektami, a okno *Project Browser* umożliwia przełączanie się między nimi.
- **5.** Prawda. Panel możesz przenieść w inne miejsce interfejsu. Wystarczy kliknąć i przeciągnąć jego zakładkę.
- **6.** Prawda. Skrót PIE pochodzi od słów *Play in Editor*. Podgląd poziomu można wyświetlić w osobnym oknie lub w wybranym panelu *Viewport*.

Ćwiczenia

Poświęć nieco czasu na zapoznanie się z interfejsem UE4. W tym ćwiczeniu utworzysz nowy poziom gry, umieścisz w nim aktorów i zapiszesz go. Są to proste, a zarazem niezbędne umiejętności i im lepiej je opanujesz, tym łatwiejsza będzie późniejsza praca z Unreal Engine.

- 1. Utwórz nowy domyślny poziom, wybierając polecenie *File/New Level* (plik/nowy poziom) lub naciskając kombinację klawiszy *Ctrl+N*.
- **2.** Skorzystaj z zakładki *Place* w panelu *Modes*, aby umieścić w poziomie aktora typu *Point Light*.
- **3.** Umieść w poziomie zasób typu *Static Mesh*, czyli siatkę statyczną. Tego typu zasoby znajdziesz w folderze *StarterContent* w panelu *Content Browser*.
- **4.** Aby zapisać poziom, kliknij prawym przyciskiem myszy folder *Content* w panelu *Content Browser* i wybierz polecenie *Add New Folder* (dodaj nowy folder), a następnie nazwij nowy folder *Maps*.
- **5.** Zapisz poziom w nowym folderze *Maps*, wybierając polecenie *File/Save* (plik/zapisz).

Skorowidz

Α

akcelerometr, 441 aktor fizyczny, 236, 280 siatki statycznej, 232 siatki szkieletowej, 185, 200 typu Ambient Sound, 122, 279 Box Trigger, 268 Camera, 217 Fog, 161 Force, 239 Matinee, 205 Physics Constraint, 236 Physics Thruster, 239 Post Processing Volume, 162 Radial Force, 240 Reflection Capture, 161 Sound, 122 Static Mesh, 78 aktywacja systemu cząstek, 182 zdarzeń, 280 albedo, 103 anatomia poziomu, 149 animacja, 188, 200 Animation, 191, 196 aplikacja Swarm Agent, 95 automatyczna ekspozycja, 435

В

bloki, 152 Blueprint klasy, 285 skrypty, 245, 267 błąd przetwarzania zawartości, 419 budowanie oświetlenia, 94 świata, 147, 148 bufory renderowania HDR, 436

С

chropowatość, 103 ciało fizyczne, 226 miękkie, 226 sztywne, 226 cienie, 88 dynamiczne, 97 Convert Scene, 192 czasomierz gry, 334 cząstki, 167

D

definiowanie siatek szkieletowych, 185 dekompozycja wypukła, 75 diagram materiału, 432 zdarzeń, 250 zdarzeń UMG, 407 diffuse, 103 dodawanie klasy Blueprint, 286 komponentów, 289 komponentów siatki statycznej, 311, 313 dokumentacja online, 35 dołączanie, 62 aktorów fizycznych, 236 dziedziczenie, 351 dźwięk, 119, 157, 160 dżojstik, 438

Ε

edytor Blueprint, 245, 247 Matinee, 207 Persona, 193 edytowanie odpowiedzi na kolizje, 81 siatek statycznych, 68, 73 efekty, 167 elementy docelowe funkcji, 273 typu Trigger Volume, 268 emitery cząstek, 168

F

flagi na liście Show, 30 folder projektu, 37 BP_Common, 337 BP_Levers, 341 BP_Pickup, 340 BP_Respawn, 340 **BP_Turrets**, 339 Content, 39 Saved, 44 funkcja, 254 Acceleration, 442 Ambient Occlusion, 435 Anti-Aliasing Method, 435 Auto Exposure, 435 Bloom, 436 Blueprint, 254 Gravity, 442 Heal Damage, 384 Lens Flares, 435 Mobile HDR, 436 Motion Blur, 435 Play Sound at Location, 278 Print String, 256

Rotation Rate, 442 Separate Transparency, 435 Spawn, 387 Tilt, 442 funkcje niestandardowe, 255 renderowania, 435

G

gęstość, 226 głowica odtwarzania, 209 Graph, 199 grupa, 209 Director, 219 grupowanie, 60

Η

HUD, 46

I

identyfikowanie wymagań, 346 importowanie plików dźwiękowych, 120 siatek statycznych, 70 siatek szkieletowych, 188 tekstur, 107 zawartości, 39 impuls, 226 instalowanie programu Launcher, 18 Unreal, 18 Unreal Engine, 19 instancje, 112 interakcje, 333 interfejs dotykowy, 436 edytora Blueprint, 248, 287 HUD, 334 UMG, 396 użytkownika, 22 interpolacja, 214

J

jednostki siatki, 55

Κ

kamera, 217 o stałej pozycji, 360 kanał zielony, 105 klasa Blueprint, 155, 298 Character, 45 DefaultPawn, 355 GameMode, 45 HUD, 46 Pawn, 45 Player Controller, 410 Point Light, 298 StartMenuController, 409 StartMenuGameMode, 409 klasy bazowe, 373 Blueprint, 285 kontrolerów, 45 potomne, 373 klipy filmowe, 205 klucz, 173 kolizje, 76, 83 aktora, 269 kolor bazowy, 103 cząstki, 179 cienia, 160 komentarze, 260 węzłów, 261 komponent Canvas, 403 komponenty, 288 aktora, 273 dźwięku, 119 siatki statycznej, 311, 313 konfigurowanie akcji wejścia, 356 bazowej klasy przeszkody, 368 kamery, 360 materiałów dla cząstek, 178

obiektu Blueprint, 323 stanu uszkodzenia, 375 kontrolki okna Curve Editor, 173 kontrolowanie dźwięków, 127 ruchu obiektu, 353 krajobrazy, 133 krzywe, 173 osi czasu, 295

L

lista Brush, 138 Falloff, 138 Tool, 138

Ł

łączenie aktorów, 155

Μ

malowanie, 139 mapa gry, 411 oświetlenia, 71 mapowanie osi, 356 Tilt, 441 mapy wysokości, 134 masa, 226 Materials, 192 materiały, 101 dla cząstek, 178 fizyczne, 231 krajobrazu, 139 menu, 411 kontekstowe Blueprint, 251 Settings, 428 Start, 401 Mesh, 191, 195 metoda wygładzania, 435 migrowanie zawartości, 41 mipmapowanie, 105 mobilność, 96

modulacja, 124 moduł, 170 Color Over Life, 176 Const Acceleration, 177 Inherit Parent Velocity, 177 Initial Color, 176 Initial Rotation, 178 Initial Size, 176 Initial Velocity, 177 Lifetime, 176 Required, 174 Rotation Rate, 178 Scale Color/Life, 176 Size By Life, 176 Spawn, 175 możliwości postaci, 334

Ν

narracja środowiska, 148 narzędzia krajobrazów, 133 przyciągania do siatki, 154 transformacji, 52 narzędzie Cascade, 168 Matinee, 209 nawigacja, 396 po scenie, 30 normalne, 104

0

obiekt typu Pawn, 374 zniszczalny, 226 obiekty do zebrania, 367 obracanie, 54 obsługa zdarzeń klawiatury, 319 ograniczanie zmiennych edytowalnych, 315 okluzja otoczenia, 435 okno Browse For Folder, 418 Curve Editor, 172, 213 edytora siatek statycznych, 68

Project Browser, 20 Reference Viewer, 43 określanie jednostek, 55 opcja Lightmass Importance Volume, 160 operatory, 259 opracowywanie zawartości, 415 optymalizacja, 430 organizacja skryptu, 260 oś czasu, 294 oświetlenie, 87, 157, 158 bezpośrednie, 87 dynamiczne, 88 kierunkowe, 91 nieba, 92 niebezpośrednie, 87 odbite, 87 punktowe, 88 reflektorowe, 90 statyczne, 88, 95 właściwości, 92

Ρ

pakiet wykonywalny, 415 dla systemu Android, 421 dla systemu iOS, 421 dla systemu Windows, 416 ustawienia zaawansowane, 421 panel Content Browser, 26, 107 Details, 25, 108 Graph, 109, 124 Landscape, 134 Message Log, 419 Modes, 23 My Blueprint, 250 Palette, 109, 124 Project Settings, 434 Tracks, 208 Viewport, 27, 108, 289 World Outliner, 25, 58 pasek menu, 23 narzędzi edytora Blueprint, 249 narzędzi edytora poziomu, 31

PBR, Physically based rendering, 102 **PIE**, 36 pierwszy projekt, 20 platforma Gameplay, 35, 44 Play in Editor, 36 pliki dźwiękowe, 120 tekstur, 106 pole komentarza, 261 polecenie listtextures, 432 Stat RHI, 434 połączenia, 251 pomiar, 55 ponowne uruchomienie gry, 378 poruszanie obiektem, 358 postacie, 345 prawa fizyki w grach, 225 predefiniowane ustawienia kolizji, 81 proces budowania świata, 149 program Launcher, 18 projekty dla urządzeń mobilnych, 425 przestrzeń dźwiękowa, 128 przesunięcie, 52 przeszkody, 367 przyciąganie do siatki, 56 przycisk Play Cue, 125 Play Node, 125 przydzielanie aktorów do zdarzeń, 270 materiału do siatki, 72 materiału fizycznego, 232, 234 pulsujące światło, 298 punkty kontrolne, 400 obrotu, 70

R

referencje do materiału, 80 zasobów, 43 renderowanie, 87, 102 efektu flary, 435 przezroczystych materiałów, 435 restytucja, 226 roślinność, 142 rozdzielczość ekranu, 399 rozmiar ekranu, 399 tekstury, 105 rozmieszczanie aktorów, 323, 325, 386 obiektów, 319 rozmycie ruchu, 435 rozpoznawanie uszkodzenia, 375 ruch przeszkody, 370 rzeźbienie kształtów, 137

S

scena, 57 sekcja Collision Responses, 83 sekcje narzędzia Cascade, 169 sekwencja animacji, 188 klasy Player, 410 zdarzeń, 272 siatka kolizji, 73, 76 obracania, 57 skalowania, 57 statyczna, 67 szkieletowa, 185, 187 siła, 226 katowa, 226 liniowa, 226 skalowanie, 53 DPI, 400 Skeletal, 190 Skeleton, 191, 194 skinning, 186 skrypt Blueprint, 182, 188 Blueprint poziomu, 267 konstrukcyjny, 307, 310 stany interakcji podczas kolizji, 83 Static Mesh, 67 struktura foldera projektu, 37 Swarm Agent, 95 symulowanie właściwości fizycznych, 229 system dźwięków, 119 menu, 411 ponownego rozmieszczania, 334 wejścia, 345 szkielet, 152, 188

Ś

ścieżka, 210, 295 ruchu, 214 typu Sound, 216 grupy Director, 219 światło kierunkowe, 91 nieba, 91

Т

tarcie, 226 tekstury, 105 SubUV, 179 testowanie gry, 32, 161, 425 tłumienie, 122, 226 Transform, 192 transformacja obrót, 54 przesunięcie, 52 skalowanie, 53 transformacje interaktywne, 54 lokalne, 54 na poziomie świata, 54 ręczne, 54 tryb Animation, 196 Designer, 396, 404 Graph, 199, 397 Mesh, 195 Skeleton, 194 tryby gry projektu, 333 interpolacji, 214 widoku, 29

tworzenie bazowej klasy przeszkody, 368 efektów, 167 funkcji Heal Damage, 384 funkcji Spawn, 387 interakcji, 333 klasy Blueprint, 320 klipów filmowych, 205 krajobrazów, 133, 135 materiałów dla elementów-zdobyczy, 381 materiału, 107, 110 menu Start, 401 niestandardowego trybu gry, 348 obiektu rozmieszczającego aktorów, 386 pakietu wykonywalnego, 415 pierwszego projektu, 20 projektu gry, 346 roślinności, 142 skryptów, 245 skryptów Blueprint, 291 skryptu funkcjonalności, 406 skryptu pulsującego światła, 298 skryptu zdarzenia, 406 stanu śmierci, 379 strzelanki platformowej, 345, 367 sygnałów dźwiękowych, 124 szkieletów i bloków, 152 świata poza poziomem, 157 tekstur, 105 unikalnych przeszkód, 369 uzdrawiających obiektów, 380 widżetu Blueprint, 395 własnego kontrolera, 351 zmiennych edytowalnych, 308 typy danych, 167 emiterów cząstek, 168 materiałów, 103 oświetlenia, 88 widoków, 28 zasobów, 42

zmiennej, 257

U

ubranie, 226 układ UV, 71 UMG, 395 Unreal Engine 4, 17 urządzenia mobilne, 426 optymalizacja, 430 podgląd gry, 426 stany ruchu, 442 ustawianie długości sekwencji, 208 platform edytora, 433 rozdzielczości, 398 tłumienia, 122 właściwości aktora, 273 ustawienia kolizji, 81 kolizji aktora, 269 mobilności, 78 pakietów, 421 projektu i fizyki, 227 siatki statycznej, 77 ustawienie Auto Activate, 182 Collision Enabled, 82 Object Type, 83 usuwanie starych przeszkód, 391 uszkadzanie obiektu, 374 używanie aktorów fizycznych, 280 edytora Blueprint, 247 edytora Persona, 193 emiterów, 170 klas Blueprint, 285, 336 komponentów, 291 materiałów, 101 materiałów fizycznych, 231 modułów, 170 skryptu konstrukcyjnego, 310, 320 zdarzeń wejścia, 358

W

warstwy, 61 wektory, 110 węzeł, 109, 251 Cast To, 376 Get Input Motion State, 442 Quality Switch, 428 Timeline, 294 zdarzeń InputTouch, 439 węzły typu Reroute, 262 wartości, 110 widżet Blueprint, 395 transformacji, 53 wirtualne dżojstiki, 437 wizualizator, 29 kanałów, 173 właściwości, 173 wizualny system skryptowy, 245 właściwości aktora, 273 aktora Matinee, 207 fizyczne, 229, 230 klasy StartMenuController, 409 klasy StartMenuGameMode, 409 kolizji aktora, 269 modulacji, 124 modułu, 171 oświetlenia, 92 światła, 93 zmiennych, 316 właściwość Activate, 276 Exposed on Spawn, 322 World Outliner, 58 wprawianie przeszkody w ruch, 370 współrzędne kartezjańskie, 51 wtyczki wykonywalne, 251 zwykłe, 251

wyrażenia warunkowe, 259 wyświetlanie siatek kolizji, 73 układów UV, 71 wyznaczanie skali, 150 zakresu, 151 wyzwalanie systemów cząsteczek, 182

Ζ

zaawansowane sygnały dźwiękowe, 127 zarządzanie krajobrazem, 136 zasoby, 42 modularne, 155 typu Matinee Data, 220 typu Physical Material, 232 typu Sound Wave, 122 typu Static Mesh, 67 zdarzenia, 252 dotyku, 439 klawiatury, 319 UMG, 407 zdarzenie Event Tick, 259 OnActorBeginOverlap, 270, 271 OnActorEndOverlap, 270, 271 OnActorHit, 270 zmiana referencji siatki, 79 zmienne, 256 edytowalne, 307 znaczniki aktora, 342 komponentu, 342

ż

żyroskop, 441

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION

1. ZAREJESTRUJ SIĘ 2. prezentuj książki 3. zbieraj prowizję

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj! http://program-partnerski.helion.pl



Jeśli dużo grasz na różnych urządzeniach, na pewno masz też pomysły na nowe gry. Jeżeli zastanawiasz się, w jaki sposób taką grę napisać, bez obaw — nie trzeba do tego magicznych umiejętności. Wystarczy wykorzystać odpowiednie narzędzie, takie jak Unreal Engine: potężny silnik gier wykorzystywany przez wielu profesjonalnych i niezależnych twórców, udostępniający szereg niezwykle przydatnych edytorów i narzędzi.

Niniejsza książka jest przeznaczona dla osób, które chcą poznać podstawy Unreal Engine 4. Ten praktyczny przewodnik składa się z 24 rozdziałów. Przestudiowanie każdego z nich wymaga najwyżej jednej godziny. W książce znalazły się szczegółowe instrukcje, praktyczne przykłady oraz liczne ćwiczenia i wskazówki. Przedstawiono istotne informacje o platformie Gameplay i opisano, jak korzystać z jednostek i systemów kontroli Unreal, aby już po kilku godzinach nauki tworzyć w pełni funkcjonalne gry na różne platformy, w tym gry internetowe i dla urządzeń mobilnych.

W tej książce między innymi:

- organizacja projektów i przepływ pracy w Unreal Engine 4
- pisanie gier planszowych
- tworzenie krajobrazów, budynków i postaci za pomocą różnych edytorów
- korzystanie z wizualnego systemu skryptów Blueprint
- tworzenie zdarzeń reagujących na akcje użytkownika
- optymalizacja gier dla urządzeń mobilnych i przygotowanie gry do wdrożenia

Aram Cookson wykłada w Savannah College of Art and Design. Od 15 lat naucza zasad grafiki i projektowania gier. Specjalizuje się w technologii Unreal Engine.

Ryan DowlingSoka jest jednym z autorów gry *Gears of War*. Należy do najbardziej doświadczonych użytkowników Unreal Engine 4.

Clinton Crumpler jest starszym grafikiem środowiskowym w Microsoft Studios. Wcześniej pracował jako grafik w BattleCry Studios. Specjalizuje się w grafice środowiskowej.



sięgnij po WIĘCEJ

Unreal Engine — świetne narzędzie dla najlepszych twórców gier!

H	elion V .	SAMS	
	księgarnia internetowa	Helion SA	
Ř	http://helion.pl	ul. Kościuszki 1c, 44-100 Gliwice tel.: 32 230 98 63 e-mail: helion@helion.pl	
	zamówienia telefoniczne http://helion.pl		KOD KORZISCI
S,	0 801 339900	Sprawdź najnowsze promocje: ● http://helion.pl/promocje Książki najchętniej czytane:	ISBN 978-83-283-3084-9
	0 601 339900	 http://helion.pl/bestsellery Zamów informacje o nowościach: http://helion.pl/nowosci 	9 788328 330849
	Informatyka w naj	ilepszym wydaniu	cena: 69,00 zł