

Inhalt

Vorwort	XVII
1 Einleitung	1
1.1 Multiplattform-Publishing	1
1.2 Das kann Unity (nicht)	2
1.3 Lizenzmodelle	2
1.4 Aufbau und Ziel des Buches	3
1.5 Weiterentwicklung von Unity	4
1.6 Online-Zusatzmaterial	4
2 Grundlagen	5
2.1 Installation	5
2.2 Oberfläche	5
2.2.1 Hauptmenü	7
2.2.2 Scene View	8
2.2.3 Game View	10
2.2.4 Toolbar	11
2.2.5 Hierarchy	13
2.2.6 Inspector	15
2.2.7 Project Browser	18
2.2.8 Console	20
2.3 Das Unity-Projekt	20
2.3.1 Neues Projekt anlegen	21
2.3.2 Bestehendes Projekt öffnen	22
2.3.3 Projektdateien	22
2.3.4 Szene	23
2.3.5 Game Objects	24
2.3.6 Components	24
2.3.7 Tags	25
2.3.8 Layer	26

2.3.9 Assets	26
2.3.10 Frames	30
2.4 Das erste Übungsprojekt	30
3 C# und Unity	33
3.1 Die Sprache C#	33
3.2 Syntax	34
3.3 Kommentare	35
3.4 Variablen	35
3.4.1 Namenskonventionen	35
3.4.2 Datentypen	36
3.4.3 Schlüsselwort var	37
3.4.4 Datenfelder/Array	37
3.5 Konstanten	39
3.5.1 Enumeration	39
3.6 Typkonvertierung	40
3.7 Rechnen	40
3.8 Verzweigungen	41
3.8.1 if-Anweisungen	41
3.8.2 switch-Anweisung	44
3.9 Schleifen	45
3.9.1 for-Schleife	45
3.9.2 Foreach-Schleife	46
3.9.3 while-Schleife	46
3.9.4 do-Schleife	46
3.10 Klassen	47
3.10.1 Komponenten per Code zuweisen	48
3.10.2 Instanziierung von Nichtkomponenten	48
3.11 Methoden/Funktionen	48
3.11.1 Werttypen und Referenztypen	50
3.11.2 Überladene Methoden	50
3.12 Lokale und globale Variablen	51
3.12.1 Namensverwechslung verhindern mit this	51
3.13 Zugriff und Sichtbarkeit	51
3.14 Statische Klassen und Klassenmember	52
3.15 Parametermodifizierer out/ref	53
3.16 Array-Übergabe mit params	54
3.17 Eigenschaften und Eigenschaftsmethoden	55
3.18 Vererbung	56
3.18.1 Basisklasse und abgeleitete Klassen	56
3.18.2 Vererbung und die Sichtbarkeit	57
3.18.3 Geerbte Methode überschreiben	57

3.18.4 Zugriff auf die Basisklasse	58
3.18.5 Klassen versiegeln	58
3.19 Polymorphie	59
3.20 Schnittstellen	59
3.20.1 Schnittstelle definieren	59
3.20.2 Schnittstellen implementieren	60
3.20.3 Zugriff über eine Schnittstellen	61
3.21 Namespaces	61
3.22 Generische Klassen und Methoden	62
3.22.1 List	63
3.22.2 Dictionary	63
4 Skript-Programmierung	65
4.1 MonoDevelop	65
4.1.1 Hilfe in MonoDevelop	66
4.1.2 Syntaxfehler	66
4.2 Nutzbare Programmiersprachen	67
4.2.1 Warum C#?	67
4.3 Unitys Vererbungsstruktur	68
4.3.1 Object	69
4.3.2 GameObject	69
4.3.3 ScriptableObject	69
4.3.4 Component	69
4.3.5 Transform	70
4.3.6 Behaviour	70
4.3.7 MonoBehaviour	70
4.4 Skripte erstellen	70
4.4.1 Skripte umbenennen	71
4.5 Das Skript-Grundgerüst	71
4.6 Unitys Event-Methoden	72
4.6.1 Update	72
4.6.2 FixedUpdate	73
4.6.3 Awake	73
4.6.4 Start	74
4.6.5 OnGUI	74
4.6.6 LateUpdate	75
4.7 Komponentenprogrammierung	75
4.7.1 Auf GameObjects zugreifen	76
4.7.2 GameObjects aktivieren und deaktivieren	77
4.7.3 GameObjects zerstören	78
4.7.4 GameObjects erstellen	78
4.7.5 Auf Components zugreifen	78
4.7.6 Components hinzufügen	80

4.7.7	Components entfernen	81
4.7.8	Components aktivieren und deaktivieren	81
4.8	Zufallswerte	81
4.9	Parallel Code ausführen	82
4.9.1	WaitForSeconds	83
4.10	Verzögerte und wiederholende Funktionsaufrufe mit Invoke	84
4.10.1	Invoke	84
4.10.2	InvokeRepeating, IsInvoking und CancelInvoke	84
4.11	Daten speichern und laden	85
4.11.1	PlayerPrefs-Voreinstellungen	85
4.11.2	Daten speichern	86
4.11.3	Daten laden	86
4.11.4	Key überprüfen	87
4.11.5	Löschen	87
4.11.6	Save	87
4.12	Szeneübergreifende Daten	88
4.12.1	Werteübergabe mit PlayerPrefs	88
4.12.2	Zerstörung unterbinden	89
4.13	Debug-Klasse	91
4.14	Kompilierungsreihenfolge	92
4.14.1	Programmsprachen mischen und der sprachübergreifende Zugriff	92
4.15	Ausführungsreihenfolge	93
5	Objekte in der dritten Dimension	95
5.1	Das 3D-Koordinatensystem	95
5.2	Vektoren	96
5.2.1	Ort, Winkel und Länge	97
5.2.2	Normalisieren	98
5.3	Das Mesh	98
5.3.1	Normalenvektor	99
5.3.2	MeshFilter und MeshRenderer	100
5.4	Transform	101
5.4.1	Kontextmenü der Transform-Komponente	101
5.4.2	Objekthierarchien	102
5.4.3	Scripting mit Transform	103
5.4.4	Quaternion	104
5.5	Shader und Materials	104
5.5.1	Material-Eigenschaften	105
5.5.2	Neues Material erstellen	108
5.5.3	Normalmaps erstellen	108
5.5.4	UV Mapping	110
5.6	3D-Modelle einer Szene zufügen	111

5.6.1	Primitives	111
5.6.2	3D-Modelle importieren	112
5.6.3	In Unity modellieren	113
5.6.4	Prozedurale Mesh-Generierung	114
6	Kameras, die Augen des Spielers	115
6.1	Die Kamera	115
6.1.1	Komponenten eines Kamera-Objektes	117
6.2	Kamerasteuerung	117
6.2.1	Statische Kamera	117
6.2.2	Parenting-Kamera	118
6.2.3	Kamera-Skripte	118
6.3	ScreenPointToRay	120
6.4	Mehrere Kameras	121
6.4.1	Kamerawechsel	121
6.4.2	Split-Screen	122
6.4.3	Einfache Minimap	123
6.4.4	Render Texture	125
6.5	Image Effects	125
6.6	Skybox	127
6.6.1	Skybox selber erstellen	127
7	Licht und Schatten	129
7.1	Ambient Light	129
7.2	Lichtarten	130
7.2.1	Directional Light	131
7.2.2	Point Light	132
7.2.3	Spot Light	132
7.2.4	Area Light	133
7.3	Schatten	134
7.3.1	Einfluss des MeshRenderers auf Schatten	135
7.4	Light Cookies	135
7.4.1	Import Settings eines Light Cookies	136
7.4.2	Light Cookies und Point Lights	136
7.5	Light Halos	138
7.5.1	Unabhängige Halos	138
7.6	Lens Flares	139
7.6.1	Eigene Flares	139
7.7	Projector	140
7.7.1	Standard Projectors	140
7.8	Lightmapping	141
7.9	Rendering Paths	144
7.9.1	Forward Rendering	144

7.9.2	Vertex Lit	145
7.9.3	Deferred Lighting	145
8	Physik in Unity	147
8.1	Physikberechnung	147
8.2	Rigidbody	148
8.2.1	Rigidbody kennenlernen	149
8.2.2	Masseschwerpunkt	150
8.2.3	Kräfte und Drehmomente zufügen	151
8.3	Kollisionen	154
8.3.1	Collider	154
8.3.2	Trigger	158
8.3.3	Static Collider	159
8.3.4	Kollisionen mit schnellen Objekten	159
8.3.5	Terrain Collider	161
8.3.6	Layer-basierende Kollisionserkennung	161
8.3.7	Mit Layer-Masken arbeiten	161
8.4	Wheel Collider	163
8.4.1	Wheel Friction Curve	164
8.4.2	Entwicklung einer Fahrzeugsteuerung	166
8.4.3	Autokonfiguration	172
8.4.4	Fahrzeugstabilität	174
8.5	Physics Materials	175
8.6	Joints	176
8.6.1	Fixed Joint	176
8.6.2	Spring Joint	176
8.6.3	Hinge Joint	177
8.7	Raycasting	177
8.8	Character Controller	178
8.8.1	SimpleMove	179
8.8.2	Move	180
8.8.3	Kräfte zufügen	181
8.8.4	Einfacher First Person Controller	181
9	Maus, Tastatur, Touch	185
9.1	Virtuelle Achsen und Tasten	185
9.1.1	Der Input-Manager	185
9.1.2	Virtuelle Achsen	187
9.1.3	Virtuelle Tasten	187
9.1.4	Steuern mit Mauseingaben	188
9.1.5	Joystick-Inputs	188
9.1.6	Anlegen neuer Inputs	189
9.2	Achsen- und Tasteneingaben auswerten	189

9.2.1	GetAxis	189
9.2.2	GetButton	190
9.3	Tastatureingaben auswerten	190
9.3.1	GetKey	191
9.3.2	anyKey	191
9.4	Mauseingaben auswerten	192
9.4.1	GetMouseButton	192
9.4.2	mousePosition	192
9.4.3	Mauszeiger ändern	193
9.5	Touch-Eingaben auswerten	194
9.5.1	Der Touch-Typ	195
9.5.2	Input.touches	195
9.5.3	TouchCount	196
9.5.4	GetTouch	196
9.5.5	Touch-Controls	196
9.6	Beschleunigungssensor auswerten	197
9.6.1	Input.acceleration	198
9.6.2	Tiefpass-Filter	199
9.7	Steuerungen bei Mehrspieler-Games	199
9.7.1	Split-Screen-Steuerung	200
9.7.2	Netzwerkspiele	200
10	Audio	203
10.1	AudioListener	203
10.2	AudioSource	204
10.2.1	Durch Mauern hören verhindern	206
10.2.2	Sound starten und stoppen	207
10.2.3	Temporäre AudioSource	208
10.3	AudioClip	209
10.3.1	3D-Sound und Hintergrundmusik	209
10.3.2	Länge ermitteln	209
10.4	Reverb Zone	210
10.5	Filter	211
11	Partikeleffekte mit Shuriken	213
11.1	Editor-Fenster	214
11.2	Particle Effect Control	215
11.3	Numerische Parametervarianten	215
11.4	Farbparameter-Varianten	216
11.5	Default-Modul	216
11.6	Effekt-Module	217
11.6.1	Emission	218
11.6.2	Shape	218

11.6.3	Velocity over Lifetime	219
11.6.4	Limit Velocity over Lifetime	219
11.6.5	Force over Lifetime	220
11.6.6	Color over Lifetime	220
11.6.7	Color by Speed	220
11.6.8	Size over Lifetime	220
11.6.9	Size by Speed	221
11.6.10	Rotation over Lifetime	221
11.6.11	Rotation by Speed	221
11.6.12	External Forces	221
11.6.13	Collision	221
11.6.14	Sub Emitter	223
11.6.15	Texture-Sheet-Animation	223
11.6.16	Renderer	224
11.7	Partikelemission starten, stoppen und unterbrechen	226
11.7.1	Play	226
11.7.2	Stop	226
11.7.3	Pause	227
11.7.4	enableEmission	227
11.8	OnParticleCollision	227
11.8.1	GetCollisionEvents	227
11.9	Feuer erstellen	228
11.9.1	Materials erstellen	229
11.9.2	Feuer-Partikelsystem	229
11.9.3	Rauch-Partikelsystem	232
11.10	Wassertropfen erstellen	236
11.10.1	Tropfen-Material erstellen	236
11.10.2	Wassertropfen-Partikelsystem	236
11.10.3	Kollisionspartikelsystem	239
11.10.4	Kollisionsound	241
12	Landschaften gestalten	243
12.1	Was Terrains können und wo die Grenzen liegen	244
12.2	Terrainhöhe verändern	244
12.2.1	Pinsel	245
12.2.2	Oberflächen anheben und senken	245
12.2.3	Plateaus und Schluchten erstellen	246
12.2.4	Oberflächen weicher machen	247
12.2.5	Heightmaps	247
12.3	Terrain texturieren	249
12.3.1	Textur-Pinsel	250
12.3.2	Texturen verwalten	250
12.4	Bäume und Sträucher	252
12.4.1	Bedienung des Place Tree-Tools	252

12.4.2	Wälder erstellen	253
12.4.3	Mit Bäumen kollidieren	253
12.5	Gräser und Details hinzufügen	254
12.5.1	Detail-Meshs	254
12.5.2	Gräser	255
12.5.3	Quelldaten nachladen	256
12.6	Terrain-Einstellungen	256
12.6.1	Base Terrain	257
12.6.2	Resolution	257
12.6.3	Tree & Details Objects	258
12.6.4	Wind Settings	258
12.6.5	Zur Laufzeit Terrain-Eigenschaften verändern	259
12.7	Der Weg zum perfekten Terrain	260
12.8	Gewässer	261
13	Wind Zones	263
13.1	Spherical vs. Directional	264
13.2	Wind Zone - Eigenschaften	265
13.3	Frische Brise	266
13.4	Turbine	266
14	GUI	267
14.1	GUIElements	268
14.1.1	GUIElements ausrichten	268
14.1.2	GUIText positionieren	269
14.1.3	GUITexture skalieren und positionieren	269
14.1.4	Interaktivität	270
14.2	3DText	271
14.3	OnGUI-Programmierung	272
14.3.1	GUI	273
14.3.2	GUILayout	275
14.3.3	GUIStyle und GUISkin	276
14.4	uGUI	278
14.4.1	Canvas	278
14.4.2	RectTransform	281
14.4.3	uGUI-Sprite Import	285
14.4.4	Grafische Controls	286
14.4.5	Interaktive Controls	289
14.4.6	Controls designen	294
14.4.7	Animationen in uGUI	295
14.4.8	Event Trigger	296
14.5	Screen-Klasse	297
14.5.1	Schriftgröße dem Bildschirm anpassen	298

15	Prefabs	299
15.1	Prefabs erstellen und nutzen	299
15.2	Prefab-Instanzen erzeugen	299
15.2.1	Instanzen per Code erstellen	300
15.2.2	Instanzen weiter bearbeiten	300
15.3	Prefabs ersetzen und zurücksetzen	301
16	Internet und Datenbanken	303
16.1	Die WWW-Klasse	303
16.1.1	Rückgabewert-Formate	304
16.1.2	Parameter übergeben	305
16.2	Datenbank-Kommunikation	306
16.2.1	Daten in einer Datenbank speichern	306
16.2.2	Daten von einer Datenbank abfragen	307
16.2.3	Rückgabewerte parsen	309
16.2.4	Datenhaltung in eigenen Datentypen	310
16.2.5	HighscoreCommunication.cs	312
16.2.6	Datenbankverbindung in PHP	313
17	Animationen	315
17.1	Allgemeiner Animation-Workflow	316
17.2	Animationen erstellen	316
17.2.1	Animation View	317
17.2.2	Curves vs. Dope Sheet	318
17.2.3	Animationsaufnahme	318
17.2.4	Beispiel Fallgatter-Animation	322
17.3	Animationen importieren	323
17.3.1	Rig	323
17.3.2	Animationen	326
17.4	Animationen einbinden	329
17.4.1	Animator Controller	329
17.4.2	Animator-Komponente	334
17.4.3	Beispiel Fallgatter Animator Controller	335
17.5	Controller-Skripte	337
17.5.1	Parameter des Animator Controllers setzen	337
17.5.2	Animation States abfragen	338
17.5.3	Beispiel Fallgatter Controller-Skript	339
17.6	Animation Events	341
17.6.1	Beispiel Fallgatter-Bewegung	342
18	Künstliche Intelligenz	345
18.1	NavMeshAgent	346
18.1.1	Eigenschaften der Navigationskomponente	346
18.1.2	Zielpunkt zuweisen	347

18.2	NavigationMesh	347
18.2.1	Object Tab	349
18.2.2	Bake Tab	349
18.2.3	Layers Tab	350
18.3	NavMeshObstacle	350
18.4	Point & Click-Steuerung für Maus und Touch	351
19	Fehlersuche und Performance	353
19.1	Fehlersuche	353
19.1.1	Breakpoints	354
19.1.2	Variablen beobachten	355
19.1.3	Console Tab nutzen	355
19.2	Performance	356
19.2.1	Rendering-Statistik	356
19.2.2	Analyse mit dem Profiler	358
19.2.3	Echtzeit-Analyse auf Endgeräten	359
20	Spiele erstellen und publizieren	361
20.1	Der Build-Prozess	361
20.1.1	Szenen des Spiels	362
20.1.2	Plattformen	363
20.1.3	Notwendige SDKs	363
20.1.4	Plattformspezifische Optionen	364
20.1.5	Developer Builds	364
20.2	Publizieren	365
20.2.1	App	365
20.2.2	Browser-Game	366
20.2.3	Desktop-Anwendung	366
21	Beispiel-Game	369
21.1	Level-Design	370
21.1.1	Modellimport	371
21.1.2	Materials zuweisen	372
21.1.3	Prefabs erstellen	372
21.1.4	Dungeon erstellen	375
21.1.5	Dekoration erstellen	378
21.2	Inventarsystem erstellen	380
21.2.1	Verwaltungslogik	380
21.2.2	Oberfläche des Inventarsystems	386
21.2.3	Inventar-Items	388
21.3	Game Controller	393
21.4	Spieler erstellen	393
21.4.1	Lebensverwaltung	395
21.4.2	Spielersteuerung	403
21.4.3	Wurfstein entwickeln	411

21.5	Quest erstellen	416
21.5.1	Erfahrungspunkte verwalten	416
21.5.2	Questgeber erstellen	418
21.5.3	Sub-Quest erstellen	425
21.6	Gegner erstellen	430
21.6.1	Model-, Rig- und Animationsimport	430
21.6.2	Komponenten und Prefab konfigurieren	431
21.6.3	Animator Controller erstellen	432
21.6.4	NavMesh erstellen	434
21.6.5	Umgebung und Feinde erkennen	435
21.6.6	Gesundheitszustand verwalten	438
21.6.7	Künstliche Intelligenz entwickeln	442
21.7	Eröffnungsszene	449
21.7.1	Startszene erstellen	449
21.7.2	Startmenü erstellen	450
21.8	Web-Player-Anpassungen	454
21.8.1	Web-Player-Template ändern	454
21.8.2	Quit-Methode im Web-Player abfangen	455
21.9	Umstellung auf uGUI	455
21.9.1	Skriptanpassungen	456
21.9.2	Controls erstellen	459
21.10	So könnte es weitergehen	466
22	Der Produktionsprozess in der Spieleentwicklung	469
22.1	Die Produktionsphasen	469
22.1.1	Ideen- und Konzeptionsphase	470
22.1.2	Planungsphase	470
22.1.3	Entwicklungsphase	470
22.1.4	Testphase	471
22.1.5	Veröffentlichung und Postproduktion	471
22.2	Das Game-Design-Dokument	471
	Schlusswort	473
	Index	475