

C#	
using	Include namespace
class	Define name
// comment	Comment: Same line
/* comment */	Comment: Multiple lines
private, protected, public	Access specifiers
bool	Variable: true or false
int	Variable: Integer number
float	Variable: decimal number
string	Variable: Text
null	Empty nothing
+ - * / % ++ --	Math functions
if(<i>condition</i>){ <i>code</i> }...else if(<i>condition</i>){ <i>code</i> }...else { <i>code</i> }	Conditional statements
== != > <	Relational operators
&&	Logical operators
switch(){case <i>var</i> : <i>code</i> break; default: <i>code</i> break;}	Similar to if conditional statement

C# (cont)	
Expression ? true: false;	Similar to if
while(<i>condition</i>){ <i>code</i> }	Loop: while
for(<i>init</i> ; <i>condition</i> ; <i>increment</i>) { <i>code</i> }	Loop: for
do{ <i>code</i> }while(<i>condition</i>)	Loop: do...while
int[] vars = new int[10];	Array wich contains 10 integers
vars[0] = 2;	Assign to an index of an array
float[] vars = new int[2]{10f, 20f};	Another array
Methods	
public void DoSmth() { <i>code</i> }	Function definition
private void Add(int num1, int num2){ <i>code</i> }	Function with parameters
private int Sum(){ <i>code</i> return <i>int</i> ;	Function with return type

MonoBehaviour	
Awake()	Called once before everything else
OnEnable()	When a GameObject becomes active
Start()	Called once after the Awake, before the Update
Update()	Called every frame. Use Time.deltaTime for time relevant code
FixedUpdate()	Called every fixed framerate frame. Used when dealing with Rigidbody.
Instantiate	Creates a copy of the original object
Destroy	Removes a gameobject, or component.
GetComponent <Type>()	Returns the component of <i>Type</i> if the game object has one attached
gameObject.transform.position	Vector3 Position of a GameObject
gameObject.transform.rotation	Quaternion rotation of a GameObject
Quaternion.Euler	Returns the quaternion of euler rotation



Other Components

Collider

`OnCollisionEnter(2D)` Called when this collider/rigidbody has begun touching another rigidbody/collider

`OnCollisionExit(2D)` Called when this collider/rigidbody has stopped touching another rigidbody/collider.

`OnTriggerEnter(2D)` Called when this Collider other enters a trigger Collider.

`OnTriggerExit(2D)` Called when this Collider other has stopped touching a trigger collider.

`isTrigger` Triggers collision without physics

`Physics.Raycast` Checks if there are colliders in a line

RigidBody

`isKinematic` Controls whether physics affects the rigidbody.

`useGravity` Controls whether gravity affects this rigidbody.

`AddForce` Applies a force of a vector to the rigidbody

Other Components (cont)

`AddTorque` Adds torque to the rigidbody

Other

`Input.GetKeyDown("Fire1")` Returns true or false for key

`Lerp` Linear interpolation between values in time from 0 to 1

`Mathf` Math functions

C

By **konsnos** (konsnos)
cheatography.com/konsnos/

Published 4th November, 2016.
Last updated 4th November, 2016.
Page 2 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>