Main Objects (Accessed via local reference only)

Game (via game)	*new* Game(width[800], height[600], renderer[Phaser.AUTO], parent[‘’], state[null], transparent[false], antialias[true], physicsConfig[null])
World (via world)	*new* World(**game**)
Camera (via camera)	*new* Camera(**game, id, x, y, width, height**)
Stage (via stage)	*new* Stage(**game**)

Game Scaling

ScaleManager (via scale)	new ScaleManager(**game, width, height**)
FlexGrid (via scale.grid)	new FlexGrid(**manager, width, height**)
FlexLayer	new FlexLayer(**manager, position, bounds, scale**)

Game Objects

GameObjectFactory (via add)	new GameObjectFactory(**game**)
GameObjectCreator (via make)	new GameObjectCreator(**game**)
Group	new Group(game, parent[game world], name['group‘], addToStage[false], enableBody[false], physicsBodyType[0]
InputHandler (via *object*.input)	new InputHandler(**sprite**)
Events (via *objects*.events)	new Events(**sprite**)
Create (via create)	new Create(**game**)

Tweens

TweenManager (via tweens)	new TweenManager(**game**)
Tween	new Tween(target, game, manager)
TweenData	new TweenData(**parent**)
Easing	new Easing() *Back, Bounce, Circular, Cubic, Elastic, Exponential, Linear, Quadratic, Quartic, Quintic, Sinusoidal*

Required
Optional
[default]
Repeatable^n

Game States

StateManager (via state)	new StateManager(**game**, pendingState[null])
State	new State()

Loader

Cache (via cache)	new Cache(**game**)
Loader (via load)	new Loader(**game**)
LoaderParser	new LoaderParser()

Graphics

Graphics	new Graphics(**game**, x[0], y[0])
BitmapData	new BitmapData(**game**, **key**, width[256], height[256])
RenderTexture	new RenderTexture(**game**, **key**, width[100], height[100], key[''], scaleMode[Phaser.scaleModes.DEFAULT], resolution [1])

Tilemaps

Tilemap	new Tilemap(**game**, key, tileWidth[32], tileHeight[32], width[10], height[10])
TilemapLayer	new TilemapLayer(**game**, **tilemap**, **index**, **width**, **height**)
Tileset	new Tileset(**name**, **firstgid**, width[32], height[32], margin[0], spacing[0], properties[{}])
Tile	new Tile(**layer**, **index**, **x**, **y**, **width**, **height**)
TilemapParser	new TilemapParser()

Sound

SoundManager (via sound)	new SoundManager(**game**)
Sound	new Sound(**game**, **key**, volume[1], loop[false])
AudioSprite	new AudioSprite(**game**, **key**)

Animation

AnimationManager (via *sprite*.animations)	new AnimationManager(sprite)
Animation	new Animation(**game**, **parent**, **name**, **frameData**, **frames**, frameRate[60], loop[false])
AnimationParser	new AnimationParser()
FrameData	new FrameData()
Frame	new Frame(**index**, **x**, **y**, **width**, **height**, **name**)

Gamepads

Gamepad (via input.gamepad)	new Gamepad(**game**)
SinglePad (via input.gamepad.pad<1,4>)	new SinglePad(**game**, **padParent**)
DeviceButton	new DeviceButton(**parent**, **buttonCode**)

System

Canvas	new Canvas()
Device (via game.device)	<internal> new Device()
DOM	new DOM()
RequestAnimationFrame (via game.raf)	new RequestAnimationFrame(**game**, forceSetTimeOut[false])

Utils

ArraySet	new ArraySet(list[new array])
ArrayUtils	new ArrayUtils()
Color	new Color()
Debug (via game.debug)	new Debug(**game**)
LinkedList	new LinkedList()
Utils	new Utils()

Time

Time (via time)	new Time(**game**)
Timer (via time.create)	new Timer(**game**, autoDestroy[true])
TimerEvent (via time.add)	new TimerEvent(**timer**, **delay**, **tick**, **repeatCount**, **loop**, **callback**, **callbackContext**, **arguments**)

Network

Net (via net)	new Net(**game**)

Math

Math (via math)	new Math()
QuadTree	new QuadTree(**x**, **y**, **width**, **height**, maxObjects[10], maxLevels[4], level[0])
RandomDataGenerator (via rnd)	new RandomDataGenerator(seeds)

Plugins

PluginManagers (via plugins)	new PluginManager(**game**)
Plugins	new Plugin(**game**, **parent**)

Signals

Signal	new Signal()
SignalBinding	new SignalBinding(**signal**, **listener**, **isOnce**, listenerContext[null], priority, args[none]^n)

v2.4.4 Cheat Sheet
Updated: 12/30/2015
Created By: @cjadelstad

Input

Input (via input)	new Input(game)
Pointer (via input.pointer)	new Pointer(**game**, **id**, pointerMode[CURSOR | CONTACT])
DeviceButton (via pointer.leftButton)	new DeviceButton(**parent**, **buttonCode**)
Keyboard (via input.keyboard)	new Keyboard(game)
Key	new Key(**game**, **keycode**)
Mouse (via input.mouse)	new Mouse(**game**)
MSPointer (via input.mspointer)	new MSPointer(new MSPointer(game))
Touch (input.touch)	new Touch(**game**)

Display

Sprite	new Sprite(**game**, **x**, **y**, **key**, **frame**)
Image	new Image(**game**, x[0], y[0], key, frame)
TileSprite	new TileSprite(**game**, **x**, **y**, **width**, **height**, **key**, **frame**)
Button	new Button(**game**, x[0], y[0], key, callback, callbackContext, overFrame, outFrame, downFrame, upFrame)
SpriteBatch	new SpriteBatch(**game**, **parent**, name[group], addToStage[false])
Rope	new Rope(**game**, **x**, **y**, **key**, **frame**, **points**)

Text

Text	new Text(**game**, **x**, **y**, **text**, style) *See compatibility table for available fonts *See API for Style properties available
BitmapText	new BitmapText(**game**, **x**, **y**, font, text[''], size[32], align['left'])
RetroFont	new RetroFont(**game**, **key**, **characterWidth**, **characterHeight**, **chars**, charsPerRow, xSpacing[0], ySpacing[0], xOffset[0], yOffset[0])

Particles

Particles (via particles)	new Particles(**game**)
Emitter	new Emitter(**game**, x[0], y[0], maxParticles[50])
Particle	new Particle(**game**, **x**, **y**, **key**, **frame**)

Physics

Physics	new Physics(**game**, physicsConfig[null])

Arcade Physics

Arcade (via physics.arcade)	new Arcade(**game**)
Body (via *sprite*.body):	new Body(**sprite**)

Ninja Physics

Ninja (via physics.ninja)	new Ninja(**game**)
Body (via *sprite*.body)	new Body(**system**, **sprite**, type[1], id[1], radius[16], x[0], y[0], width[0], height[0])
AABB	new AABB(**body**, **x**, **y**, **width**, **height**)
Circle	new Circle(**body**, **x**, **y**, **radius**)
Tile	new Tile(**body**, **x**, **y**, **width**, **height**, type[1])

P2 Physics

P2 (via physics.p2)	new P2(**game**, config)
Body (via *sprite*.body)	new Body(**game**, sprite, x[0], y[0], mass[1])
BodyDebug	new BodyDebug(**game**, **body**, **settings**)
Material	new Material(**name**)
ContactMaterial	new ContactMaterial(**materialA**, **materialB**, options)
CollisionGroup	new CollisionGroup(**bitmask**)
FixtureList	new FixtureList(**list**)
Constraints	
Distance Constraint	new DistanceConstraint(**world**, **bodyA**, **bodyB**, **distance**, localAnchorA, localAnchorB, maxForce[Number.MAX_VALUE])
GearConstraint	new GearConstraint(**world**, **bodyA**, **bodyB**, angle[0], ratio[1])
LockConstraint	new LockConstraint(**world**, **bodyA**, **bodyB**, offset, angle[0], maxForce)
PrismaticConstraint	new PrismaticConstraint(**world**, **bodyA**, **bodyB**, lockRotation[true], anchorA, anchorB, axis, maxForce)
RevoluteConstraint	new RevoluteConstraint(**world**, **bodyA**, **pivotA**, **bodyB**, **pivotB**, maxForce[0], worldPivot[null])
PointProxy	new PointProxy(**world**, **destination**)
InversePointProxy	new InversePointProxy(**world**, **destination**)
Spring	new Spring(**world**, **bodyA**, **bodyB**, restLength[1], stiffness[100], damping[1], worldA, worldB, localA, localB)
RotationalSpring	new RotationalSpring(**world**, **bodyA**, **bodyB**, restAngle, stiffness[100], damping[1])

Geometry

Circle	new Circle(x[0], y[0], diameter[0])
Ellipse	new Ellipse(x[0], y[0], width[0], height[0])
Line	new Line(x1[0], y1[0], x2[0], y2[0])
Point	new Point(x[0], y[0])
Polygon	new Polygon(**points**)
Rectangle	new Rectangle(**x**, **y**, **width**, **height**)
RoundedRectangle	new RoundedRectangle(x[0], y[0], width[0], height[0], radius[0])

Required
Optional
[default]
Repeatable^n

v2.4.4 Cheat Sheet
Updated: 12/30/2015
Created By: @cjadelstad