

SSC BuiltIn and standard utility documentation:

I don't know exactly what all these functions do and there may be mistakes. If you would like to contribute some knowledge, please DM me on discord Augs#3262 and I can give you access.

If a parameter has a generic name like "a" or "b" then that means I don't know what it should be.

If you see something like `print(...)` then that ellipsis means that function is variadic.

If you see something like `void srand(int seed)` that means that the `seed` should be an integer and this returns `void`.

You can pass anything you want into these functions but passing the wrong type is not recommended.

A "bool" is just an int that is either 0 or not.

For more details on the language see the ReadMe: <https://github.com/arookas/ssc>

common.sun:

int

Format:

```
int int(obj x)
```

Returns x as an int

float

Format:

```
float float(obj x)
```

Returns x as a float

typeof

Format:

```
int typeof(obj x)
```

Returns type of x.

Types are:

int = 0

float = 1
string = 2

dump

Format:

```
void dump()
```

Unknown.

print

Format:

```
void print(...)
```

Prints arguments, but where?

random.sun:

srand

Format:

```
void srand(int seed)
```

Returns x as an int

rand

Format:

```
int rand()
```

Returns a random number based on the seed.

typeof

Format:

```
typeof(x)
```

Returns type of x?

dump

Format:

```
dump()
```

Unknown.

print

Format:

```
print(...)
```

Prints arguments, but where?

scene.sun:

setNextStage

Format:

```
void setNextStage(int StageID, int? b)
```

Sets the next stage that will be loaded.

getAddressFromViewObjName

Format:

```
int getAddressFromViewObjName(string ObjectName)
```

Returns the address of an object with a given name.

getNameRefHandle

Format:

```
int getNameRefHandle(string ObjectName)
```

Returns the handle of an object with a given name.

If multiple objects have the same name then I think it chooses only one instance.

Handles are unique?

getNameRefName

Format:

```
string getNameRefName(int handle)
```

Returns the name of an object with a given handle.

getNPCType

Format:

```
??? getNPCType(int NPCHandle?)
```

Unknown

setTransScale

Format:

```
void setTransScale(string ObjectName, float x, float y, float z,  
float ScaleX, float ScaleY, float ScaleZ);
```

Set's the transformation of an object. It can change Scale and position.

isWaterMelonIsReached

Format:

```
bool isWaterMelonIsReached(string MelonName?)
```

Returns true if the waterMelon has reached the right location

killMushroom1up

Format:

```
void killMushroom1up(int 1UpHandle)
```

Disables the given 1Up

checkMonteClear

Format:

```
bool checkMonteClear(string PiantaName)
```

Checks if a given pianta is clear from goop?

raiseBuilding

Format:

```
void raiseBuilding(int a)
```

Raises a given building, like the hill to the bridge in bianco hills.

startOpenModelGate

Format:

```
void startOpenModelGate(??? a)
```

Unknown.

maniCoinDown

Format:

```
??? maniCoinDown()
```

Unknown.

Cubes

isInsideCube

Format:

```
bool isInsideCube(string CubeName)
```

Returns true if Mario is in a given cube.

isInsideFastCube

Format:

```
bool isInsideFastCube(int CubeTable?, int CubeIndex?)
```

Returns true if Mario is inside a given fast cube.

There are 3 fast cube tables, I believe these cubes will push Mario, but I'm not sure.

Appear

appear8RedCoinsAndTimer

Format:

```
void appear8RedCoinsAndTimer()
```

Makes hidden red coins visible and makes the timer appear for timed red coin missions. The length of the timer is set by a parameter on the red coin switch.

appearMushroom1up

Format:

```
void appearMushroom1up(int? 1UpHandle);
```

Makes invisible 1Up appear

DummyNPC

connectDummyNpc

Format:

```
void? connectDummyNpc(int? DummyHandle?)
```

Unknown.

onTalkToDummyNpc

Format:

```
bool? onTalkToDummyNpc()
```

Returns true if you are talking to the DummyNPC?

Nerves

checkCurNerve4Npc

Format:

```
bool? checkCurNerve4Npc(??? a, ??? b)
```

Unknown.

checkLastestNerve4Npc

Format:

```
void checkLastestNerve4Npc(??? a, ??? b)
```

Unknown.

pushNerve4LiveActor

Format:

```
void pushNerve4LiveActor(??? a, ??? b)
```

Unknown.

Boxes

checkWoodBox

Format:

```
bool? checkWoodBox(??? a, ??? b)
```

Returns true if the specified box is still there?

refreshWoodBox

Format:

```
void refreshWoodBox(int BoxHandle?)
```

Respawns the specified box?

killWoodBox

Format:

```
void killWoodBox(??? a, ??? b)
```

Despawns the specified box?

Distance checks

isNearActors

Format:

```
bool isNearActors(string ActorName, float Distance, ...string  
OtherActorNames)
```

Returns true if one of the other actors is within the specified distance of the first actor.

Note: “マリオ” is Mario

isNearSameActors

Format:

```
bool isNearSameActors(??? a, ??? b)
```

Unknown.

Shines

appearShine

Format:

```
void appearShine(string Camera, string ShineName)
```

Makes a specified shine appear, viewed from the specified camera.

appearShineForWoodBox

Format:

```
void appearShineForWoodBox(int Box)
```

Makes a shine appear from a given box.

But what Shine?

appearShineFromKageMario

Format:

```
void appearShineFromKageMario(string Shine?, string Camera?, int KageMarioAddress?)
```

Makes a shine spawn from shadow Mario.

appearShineFromNpc

Format:

```
void appearShineFromNpc(string Shine, string NPCName, string Camera)
```

Makes a shine spawn from a given NPC, viewed from a specified camera.

appearShineFromNPCWithoutDemo

Format:

```
void appearShineFromNPCWithoutDemo(string Shine, string NPCName)
```

Makes a shine spawn from a given NPC, with no cutscene.

killShine

Format:

```
void killShine(string shineName)
```

Disables a given Shine.

NPCs

setCollision

Format:

```
void setCollision(string ActorName?, ??? Collision)
```

Unknown.

setFlagNPCCanTaken

Format:

```
void setFlagNPCCanTaken(string NPCName, bool Flag)
```

Set the property that enables Mario to pick up a given NPC.

setFlagNPCDead

Format:

```
void setFlagNPCDead(string NPCName, bool Flag)
```

Set the property that disables/enables the NPC.

setFlagNPCDontTalk

Format:

```
void setFlagNPCDontTalk(string NPCName, bool Flag)
```

Set the property that means the NPC no longer talks.

setFlagNPCDontThrow

Format:

```
void setFlagNPCDontThrow(string NPCName, bool Flag)
```

Set the property that enables Mario to throw a given NPC.

isOnLiveActorFlag

Format:

```
void isOnLiveActorFlag(string ActorName, bool Flag)
```

Unknown.

setHide4LiveActor

Format:

```
void setHide4LiveActor(string ActorName, bool Flag)
```

Set the given actor to hide.

setDead4LiveActor

Format:

```
void setDead4LiveActor(string ActorName, bool Flag)
```

Set the given actor to die.

setNpcBalloonMessage

Format:

```
void setNpcBalloonMessage(??? a, ??? b, ??? c)
```

Change FLUDD's next message.

setNpcTalkForbidCount

Format:

```
void setNpcTalkForbidCount(??? a, ??? b)
```

Unknown.

npcDanceOffHappyOn

Format:

```
void npcDanceOffHappyOn(string NPCName)
```

Make the given NPC stop dancing and just be happy ☺?

npcDanceOn

Format:

```
void npcDanceOn(string NPCName)
```

Make the given NPC dance?

isNpcSinkBottom

Format:

```
bool isNpcSinkBottom(string NPCName)
```

Unknown.

Mario

setMarioWaiting

Format:

```
void setMarioWaiting()
```

Makes Mario wait.

startMarioTalking

Format:

```
void startMarioTalking()
```

Unknown.

warpMario

Format:

```
void warpMario(float x, float y, float z, float Direction)
```

Warps Mario to a given location, facing a given direction.
Direction is in degrees?

warpFrontToMario

Format:

```
void warpFrontToMario(int ObjAddress)
```

Warp a given object to a couple of metres in front of Mario. The object faces Mario.

onNeutralMarioKey

Format:

```
bool? onNeutralMarioKey()
```

Returns true if Mario is waiting?

Piantissimo

startMonteman

Format:

```
void startMonteman(int PiantissimoAddress?)
```

Start the race?

monteManReachFlag

Format:

```
bool monteManReachFlag()
```

Returns true if Piantissimo has reached the flag.

Shadow Mario

startEMarioDisappear

Format:

```
void warpFrontToMario(int EMarioAddress)
```

Make Shadow Mario disappear?

startEMarioGateDrawing

Format:

```
void startEMarioGateDrawing(int EMarioAddress)
```

Make Shadow Mario open Bianco hills?

startEMarioRunAway

Format:

```
void startEMarioRunAway(int EMarioAddress)
```

Trigger Shadow Mario's pad file?

isEMarioDownWaitingToTalk

Format:

```
bool isEMarioDownWaitingToTalk(int EMarioAddress)
```

Returns true if Shadow Mario is on the ground waiting to talk.

isEMarioReachedToGoal

Format:

```
bool isEMarioReachedToGoal(int EMarioAddress)
```

Has EMario finished his pad file?

sound.sun:

startBGM

Format:

```
void startBGM(int SongID)
```

Start some background music.

stopBGM

Format:

```
void stopBGM(int SongID)
```

Stop the given background track.

startSE

Format:

```
void startSE(int SoundID)
```

Start a sound effect.

startEventSE

Format:

```
void startEventSE(int SoundID)
```

Start an event sound effect.

Not sure what the difference is from a normal sound effect.

startMontemanBGM

Format:

```
void startMontemanBGM()
```

Starts the piantissimo music.

startMontemanFanfare

Format:

```
void startMontemanFanfare()
```

Starts the piantissimo end music?

system.sun:

Flags

getSystemFlag

Format:

```
int getSystemFlag(int FlagID)
```

Returns the value of a given flag.

setSystemFlag

Format:

```
void setSystemFlag(int FlagID, int Value)
```

Set the value of a given flag.

Pollution

isGraffitoCoverage0

Format:

```
bool isGraffitoCoverage0()
```

Returns true if there is no graffito

getPollutionLevel

Format:

```
int getPollutionLevel()
```

Gets the amount of pollution in the level.

setGraffitoMultiplied

Format:

```
void setGraffitoMultiplied(??? a)
```

Unknown.

setPollutionIncreaseCount

Format:

```
void setPollutionIncreaseCount(??? count)
```

Unknown.

Event

setEventForWaterMelon

Format:

```
void setEventForWaterMelon()
```

Unknown.

setEventID

Format:

```
void setEventID(??? a, ??? b)
```

Unknown.

isMapEventFinishedAll

Format:

```
bool isMapEventFinishedAll()
```

Unknown

Demo

isDemoMode

Format:

```
bool isDemoMode()
```

Returns true if we are in a cutscene.

fireStartDemoCamera

Format:

```
void fireStartDemoCamera(??? a)
```

Plays intro again?

launchEventClearDemo

Format:

```
??? launchEventClearDemo()
```

Unknown.

startMareBottleDemo

Format:

```
void startMareBottleDemo()
```

Plays cutscene for bottle in noki bay.

isFinishMareBottleDemo

Format:

```
bool isFinishMareBottleDemo()
```

Returns true if the bottle cutscene is finished.

waitForFinishDemo

Format:

```
void waitForFinishDemo()
```

Yields until the demo has finished.

Time

setTimeLimit

Format:

```
void setTimeLimit(int TimeLimit)
```

Sets time limit for timer.

setAttentionTime

Format:

```
void setAttentionTime(int Time)
```

Unknown.

getTime

Format:

```
int getTime()
```

Returns amount of time left on timer.

getRestTime

Format:

```
void getRestTime()
```

Unknown.

insertTimer

Format:

```
void insertTimer(int mode, int TimeLeft)
```

Initialise a timer with TimeLeft number of seconds.

Modes:

0 - Unknown

1 - Unknown

2 - Countdown

startTimer

Format:

```
void startTimer()
```

Start the timer, and can only be used after a timer has been initialised.

stopTimer

Format:

```
void stopTimer()
```

Stop the timer.

Misc

invalidatePad

Format:

```
void invalidatePad(int Number)
```

Unknown.

isGameModeNormal

Format:

```
bool isGameModeNormal()
```

Returns true if the game mode is normal, i.e. Mario can walk, jump, ect...

isBossDefeated

Format:

```
bool isBossDefeated()
```

Detect if a boss has been defeated. Used for squid boss.

registerMovie

Format:

```
bool registerMovie(int MovieIdx)
```

Start a THP movie.

appearReadyGo

Format:

```
void appearReadyGo()
```

Make "Ready? Go!" appear on the screen

startAppearJetBalloon

Format:

```
void startAppearJetBalloon(??? a, ??? b)
```

Something to do with starting the roller coaster boss fight

changeNozzle

Format:

```
void changeNozzle(int NozzleType)
```

Changes FLUDD's nozzle.

Nozzle Type list:

0: Spray

1: Rocket

2: Underwater

3: Spray from spawn

4: Hover

5: Turbo

6 and up: Game crash

changeSunglass

Format:

```
void changeSunglass(int SunglassesType)
```

Changes Mario's outfit.

putNozzle

Format:

```
void putNozzle(int Nozzle)
```

Unknown.

startMiss

Format:

```
void startMiss()
```

Kills Mario

gameOver

Format:

```
void gameOver()
```

Triggers game over?

eggYoshiStartFruit

Format:

```
void? eggYoshiStartFruit(??? a)
```

Changes which fruit Yoshi wants at the start?

getFruitNum

Format:

```
int getFruitNum(??? a, ??? b)
```

Unknown.

setFruitType

Format:

```
void setFruitType(??? a, ??? b, ??? c)
```

Unknown.

resetFruitNum

Format:

```
void resetFruitNum(??? fruit)
```

Unknown.

talk.sun:

getTalkMode

Format:

```
int getTalkMode()
```

Gets the current talk mode

isTalkModeNow

Format:

```
bool isTalkModeNow()
```

Returns true if Mario is talking to someone

setTalkMsgID

Format:

```
void setTalkMsgID(int MsgID, int Flags)
```

Set the current textbox message with different flags.

Don't use this in practice, see: talkAndWait and talkAndClose

getTalkNPC

Format:

```
int getTalkNPC()
```

Get the handle of the NPC Mario is talking to.

getTalkNPCName

Format:

```
string getTalkNPCName()
```

Get the name of the NPC Mario is talking to.

setValue2TalkVariable

Format:

```
void setValue2TalkVariable(??? var, ??? value)
```

Unknown.

talkAndWait

Format:

```
void talkAndWait(int msgID)
```

Display the given message and wait indefinitely. Not recommended.

talkAndClose

Format:

```
void talkAndClose(int msgID)
```

Display the given message and let Mario close the textbox.

talkAndSelect

Format:

```
void talkAndSelect(int msgID)
```

Display the given message with a choice and wait indefinitely. Not recommended.

talkAndSelectClose

Format:

```
void talkAndSelectClose(int msgID)
```

Display the given message with a choice and let Mario close the textbox.

forceTalk

Format:

```
void forceTalk(int Handle)
```

Forces Mario to talk with a given NPC. Does not trigger textbox that must be done with talkAndClose

forceTalkExceptNpc

Format:

```
void forceTalkExceptNpc(int Handle)
```

Unknown.