Minecraft & Python

Dr W. H. Bell : http://www.whbell.net/

1) Read through the getting started information.
2) Start running Idle for Python3 and Minecraft.
3) `printPositionOnce.py` – load and run the program. Then move the player and rerun the program.
4) `addBlockOnce.py` – load and run the program. Then change the block type and rerun the program.

Challenge : raining sand

- Create `SAND` blocks above the player in several places.
- Hint – copy and paste the `setBlock` function call several times with different coordinates.

---

5) `printPosition.py` – load and run the program. Try moving the player around. This program contains a while loop and a sleep statement. Try changing the value in the sleep statement and rerun it.

Challenge : chasing blocks

- Write a program that continues to create blocks where the player is.
- Hint – use a `while` loop.

---

6) `createTriangle.py` – load and run the program. Try changing the size of the triangle, by altering the `setBlocks` function call and changing the limits of the `for` loop.

Challenge : pyramid

- Write a program to create a pyramid that is ten bricks high.
- Hint – start from the triangle example.

Challenge : bouncing brick

- Write a program where a block moves up and down.
- To do this use an `AIR` brick to perform the animation, removing the last brick before putting another brick in the next position.

---

7) `checkPosition.py` – load and run the program.
8) `changePositionOnce.py` – load and run the program. Try changing the position the player is moved to.

Challenge : monolith teleporter

- Write a program that transports the player to another place in the world when the get too close to the monolith.
- Hint – use the example programs.

Challenge : trapped

- Choose a trap area. When a player steps into the selected x-z plane, encase them in SAND.

---

9) `checkDirection.py` – load and run the program. Then try moving about the screen. This program contains a function definition. The function returns a Vec3 object, which has x, y, and z components. This is the same object as `getPos()` or `getTilePos()` returns.

Challenge : magic bridge

- Create a program that produces a bridge of ice in front of the player.

Challenge : bulldozer

- Write a program to remove blocks from in front of the player.
- Hint – use an AIR brick and the player direction.

At the end of the class, blow up the world using TNT. Place a large amount of TNT, with one active block. Then smash the active block.

The full Python application programming interface for Minecraft is described at:
http://www.stuffaboutcode.com/p/minecraft-api-reference.html