

Lesson 8

Types: Deactivating a Portal

greenPortal.isActive = false

```
func moveThree() {  
    moveForward()  
    moveForward()  
    moveForward()  
}  
for i in 1...3 {  
    moveThree()  
    turnRight()  
    moveThree()  
    toggleSwitch()  
    turnLeft()  
    turnLeft()  
}
```



Lesson 8

Types: Portal On and Off

```
func moveAndCollect() {  
    while !isBlocked {  
        moveForward()  
        if isOnGem {  
            collectGem()  
        }  
    }  
}
```

```
func turnAround() {  
    turnLeft()  
    turnLeft()  
}
```

```
moveAndCollect()  
turnAround()  
purplePortal.isActive = false  
while !isBlocked {  
    moveForward()  
}  
toggleSwitch()  
turnAround  
purplePortal.isActive = true  
moveAndCollect()
```



Lesson 8

Types: Portal On and Off

```
func moveAndCollect() {  
    while !isBlocked {  
        moveForward()  
        if isOnGem {  
            collectGem()  
        }  
    }  
}
```

```
func turnAround() {  
    turnLeft()  
    turnLeft()  
}
```

```
moveAndCollect()  
turnAround()  
purplePortal.isActive = false  
while !isBlocked {  
    moveForward()  
}  
toggleSwitch()  
turnAround  
purplePortal.isActive = true  
moveAndCollect()
```



Lesson 8

Setting the Right Portal

```
func moveCollect() {  
    moveForward()  
    collectGem()  
}
```

```
func turnAround() {  
    turnLeft()  
    turnLeft()  
}
```

```
moveForward()  
moveCollect()  
turnAround()  
bluePortal.isActive = false  
moveForward()  
moveCollect()  
turnAround()
```

```
bluePortal.isActive = true  
pinkPortal.isActive = false  
moveForward()  
moveForward()  
moveForward()  
collectGem()  
turnAround()  
pinkPortal.isActive = true  
moveForward()  
turnAround()  
moveCollect()
```



Lesson 8

Setting the Right Portal

```
func moveCollect() {  
    moveForward()  
    collectGem()  
}
```

```
func turnAround() {  
    turnLeft()  
    turnLeft()  
}
```

```
moveForward()  
moveCollect()  
turnAround()  
bluePortal.isActive = false  
moveForward()  
moveCollect()  
turnAround()
```

```
bluePortal.isActive = true  
pinkPortal.isActive = false  
moveForward()  
moveForward()  
moveForward()  
collectGem()  
turnAround()  
pinkPortal.isActive = true  
moveForward()  
turnAround()  
moveCollect()
```



Lesson 8

Types: Corners of the World

N/A

Types: Random Gems Everywhere

```
var gemCounter = 0

bluePortal.isActive = false
pinkPortal.isActive = false
while gemCounter < totalGems {
  if isOnGem() {
    collectGem()
    gemCounter = gemCounter + 1
  }
  moveForward()
  if isBlocked() {
    turnLeft()
    turnLeft()
    if bluePortal.isActive == true {
      bluePortal.isActive = false
      pinkPortal.isActive = false
    } else if bluePortal.isActive == false {
      bluePortal.isActive = true
      pinkPortal.isActive = true
    }
  }
}
```



Lesson 8

Types: Corners of the World

N/A

Types: Random Gems Everywhere

```
var gemCounter = 0

bluePortal.isActive = false
pinkPortal.isActive = false
while gemCounter < totalGems {
  if isOnGem() {
    collectGem()
    gemCounter = gemCounter + 1
  }
  moveForward()
  if isBlocked() {
    turnLeft()
    turnLeft()
    if bluePortal.isActive == true {
      bluePortal.isActive = false
      pinkPortal.isActive = false
    } else if bluePortal.isActive == false {
      bluePortal.isActive = true
      pinkPortal.isActive = true
    }
  }
}
```



Lesson 8

Initialization: Initializing Your Expert

```
let expert = Expert()
```

```
func solveSide() {  
    expert.moveForward()  
    expert.moveForward()  
    expert.moveForward()  
    if expert.isOnGem {  
        expert.collectGem()  
    } else {  
        expert.turnLockUp()  
    }  
}
```

```
func returnToCenter() {  
    expert.turnLeft()  
    expert.turnLeft()  
    expert.moveForward()  
    expert.moveForward()  
    expert.moveForward()  
    expert.turnRight()  
}
```

```
for i in 1...3 {  
    solveSide()  
    returnToCenter()  
}  
solveSide()
```



Lesson 8

Initialization: Initializing Your Expert

```
let expert = Expert()
```

```
func solveSide() {  
    expert.moveForward()  
    expert.moveForward()  
    expert.moveForward()  
    if expert.isOnGem {  
        expert.collectGem()  
    } else {  
        expert.turnLockUp()  
    }  
}
```

```
func returnToCenter() {  
    expert.turnLeft()  
    expert.turnLeft()  
    expert.moveForward()  
    expert.moveForward()  
    expert.moveForward()  
    expert.turnRight()  
}
```

```
for i in 1...3 {  
    solveSide()  
    returnToCenter()  
}  
solveSide()
```



Lesson 8

Initialization: Train Your Expert

N/A

Initialization: Using Instances of Different Types

```
let expert = Expert()  
let character = Character()
```

```
expert.moveForward()  
expert.turnLockUp()  
character.moveForward()  
character.collectGem()  
character.moveForward()  
character.turnRight()  
character.moveForward()  
character.moveForward()  
expert.turnLockDown()  
expert.turnLockDown()  
character.turnRight()  
character.moveForward()  
character.collectGem()
```



Lesson 8

Initialization: Train Your Expert

N/A

Initialization: Using Instances of Different Types

```
let expert = Expert()  
let character = Character()
```

```
expert.moveForward()  
expert.turnLockUp()  
character.moveForward()  
character.collectGem()  
character.moveForward()  
character.turnRight()  
character.moveForward()  
character.moveForward()  
expert.turnLockDown()  
expert.turnLockDown()  
character.turnRight()  
character.moveForward()  
character.collectGem()
```



Lesson 8

Initialization: It Takes 2

```
let expert = Expert()
let character = Character()
```

```
func turnCorner() {
    expert.moveForward()
    expert.moveForward()
    expert.turnRight()
    expert.moveForward()
    expert.moveForward()
}
expert.turnLeft()
expert.moveForward()
turnCorner()
expert.turnLeft()
expert.turnLockDown()
expert.turnLockDown()
expert.turnLockDown()
```

```
character.moveForward()
character.moveForward()
character.collectGem()
expert.turnRight
turnCorner()
expert.moveForward()
expert.moveForward()
turnCorner()
expert.turnLeft()
expert.turnLockUp()
character.moveForward()
character.moveForward()
character.toggleSwitch()
```

***Some puzzles may have multiple solutions**

Lesson 8

Initialization: It Takes 2

```
let expert = Expert()
let character = Character()
```

```
func turnCorner() {
    expert.moveForward()
    expert.moveForward()
    expert.turnRight()
    expert.moveForward()
    expert.moveForward()
}
expert.turnLeft()
expert.moveForward()
turnCorner()
expert.turnLeft()
expert.turnLockDown()
expert.turnLockDown()
expert.turnLockDown()
```

```
character.moveForward()
character.moveForward()
character.collectGem()
expert.turnRight
turnCorner()
expert.moveForward()
expert.moveForward()
turnCorner()
expert.turnLeft()
expert.turnLockUp()
character.moveForward()
character.moveForward()
character.toggleSwitch()
```

***Some puzzles may have multiple solutions**

