

// RANDOM

```
import random

random.randint(1, 6)    # both ends included
random.choice("Bull")  # random item from sequence
```

import goes at the top. Every run gives different results.

// WHILE – the basic loop

```
count = 0

while count < 5:
    print(count)
    count += 1    # always update

# 0 1 2 3 4
```

Runs while condition is True. You control what changes.

// WHILE WITH AND / OR / NOT

```
# and – both must be True
while answer != pwd and attempts < 3:
    ...

# or – at least one must be True
while health > 0 or ammo > 0:
    ...

# not – inverts the condition
while not done:
    ...
```

Same rules as if. Combine as needed.

// WHILE + RANDOM

```
roll = 0
attempts = 0

while roll != 6:
    roll = random.randint(1, 6)
    attempts += 1

print(f"Got 6 after {attempts} attempt(s).")
```

Generate inside the loop — fresh each iteration.

// WHILE TRUE + BREAK

```
while True:
    entry = input('Number or stop: ')
    if entry == 'stop':
        break
    total += int(entry)

# runs forever – break is the only exit
# use when exit condition lives inside
```

Every while True needs a reachable break.

// WHILE...ELSE

```
while attempts < 3:
    guess = int(input('Guess: '))
    attempts += 1
    if guess == secret:
        print('Correct!')
        break
else:
    print("Out of attempts.")

# else runs – condition became False
# else skipped – break interrupted
```

Completed or interrupted — while knows the difference.

// COMMON MISTAKES

```
! Always update inside – or loop runs forever
! Check condition before loop – might never run
! Every while True needs a reachable break
! Validate input before int() – crashes on words
! < vs <= – know which boundary you want
! else skipped when break fires
```