Definite Loops: For Loop

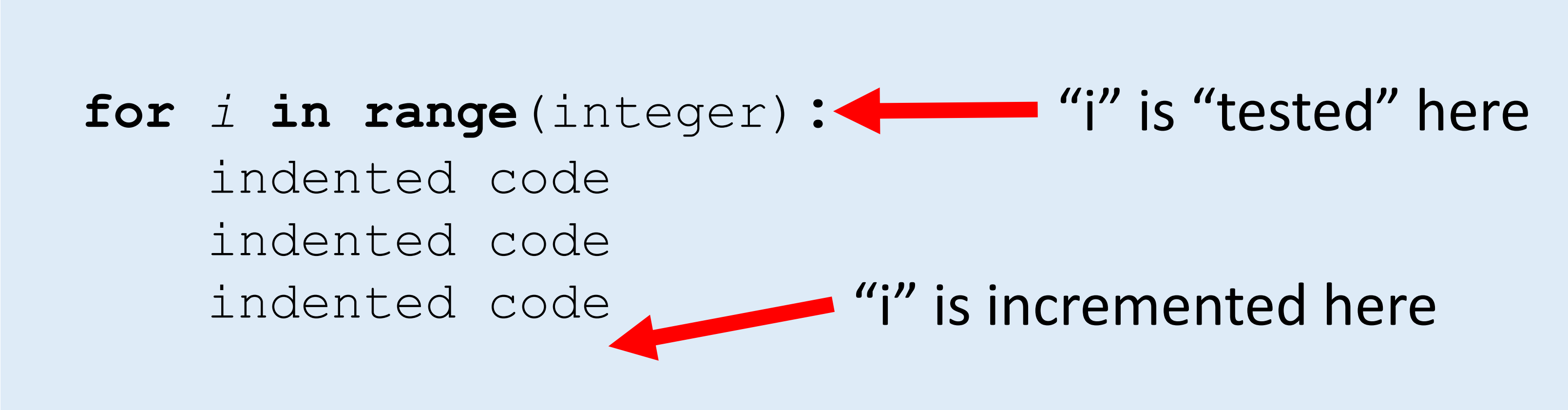
- A definite loop executes a definite number of times, i.e., at the time Python starts the loop it knows exactly how many iterations to do.

```
for itirationVariable in expresionList:
    indented code
    indented code
    indented code
```

- The beginning and end of the body are indicated by indentation.

For Loops

- When you want to repeat code "i" number of time.
- loop for "i" times, where "i" is "range(n -1)“



```
for i in range(integer):
          indented code
          indented code
          indented code
```

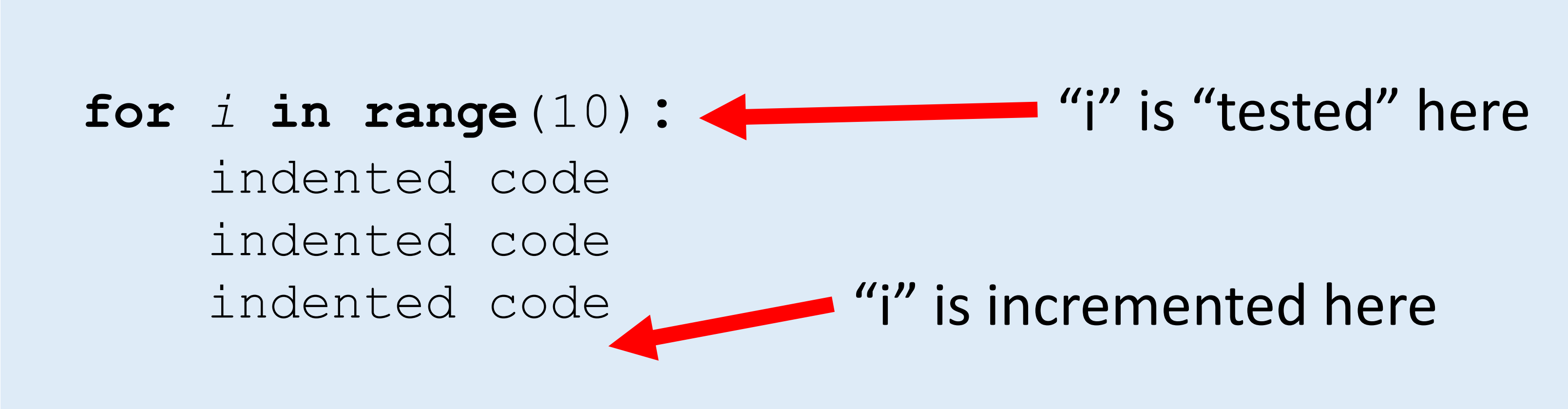
“i” is “tested” here

“i” is incremented here

- If the for loop range was (10), then the loop counter will start at 0 and loop 10 times… 0-9

For Loops

- When you want to repeat code "i" number of time.
- loop for "i" times, where "i" is "range(n -1)“



```
for i in range(10):
     indented code
     indented code
     indented code
```

“i” is “tested” here

“i” is incremented here

- If the for loop range was (10), then the loop counter will start at 0 and loop 10 times… 0-9

For Loops: Example Code

- When you want to repeat code "i" number of time.
- loop for "i" times, where "i" is "range(n -1)“

```
def main():

        for i in range(10):
                print(i)


main()
```

```
def main():

        for i in range(1,10):
                print(i)


main()
```

- If the for loop range was (10), then the loop counter will start at default 0 and loop 10 times… 0→9
- If the start is defined in the range(1,10) then the loop counter will start at default 0 and loop 9 times… 1→9

For Loops Lab #1

- Write a program that uses the for loop
  - Print the value of “I”
  - Add a variable and do some math like:

```
print(i)
sum = i*5
print(sum)
```

```
def main():

    for i in range(10):
        print(i)


main()
```

```
def main():

    for i in range(1,10):
        print(i)


main()
```

- What happens if you try to modify “I”?

```
print(i)
i=5
print(”does”,i,”= 5?”)
```

STEAM Clown™
& Squeaky Hinge
Productions

For Loops: Expresion List

- The `for loop` is also used to iterate over elements of a sequence list
- When you want to repeat code "n" number of time.
- It works like this: " for all elements in a list, do this "

```
def main():
      expresionList = ["stringA", "stringC", "stringB"]
      for targetVariable in expresionList:
            print()

main()
```

For Loops

- The `for loop` counts up the number (n) of elements in the sequence list
- The loop repeats "n" number of time.
- It works like this: " for all elements in a list, do this "

```
def main():
        computerBrands = ["Apple", "Asus", "Dell", "Samsung"]
        for brands in computerBrands:
        print(brands)

main()
```

For Loops Lab #2

- Write a program that uses the for loop based on a `expresionList`

```
def main():
      computerBrands = ["Apple", "Asus", "Dell", "Samsung"]
      for brands in computerBrands:
      print(brands)

main()
```

For Loops: Break, Continue

- To break out from a loop, you can use the keyword "break“

```
def main():

      for i in range(10):
        if i == 3:
            break
        print(i)

main()
```

```
def main():

      for i in range(10):
        if i == 3:
            continue
        print(i)

main()
```

- The continue statement is used to tell Python to skip the rest of the statements in the current loop block and to continue to the next iteration of the loop.

For Loops Lab #3

- Write a program that uses the `break` and `continue`

```
def main():

        for i in range(10):
          if i == 3:
                break
          print(i)

main()
```

```
def main():

      for i in range(10):
        if i == 3:
            print("traped",3)
            continue
        if i == 8:
            print("traped",8)
            continue
        print(i)

main()
```

- Why would you use these commands?