```
>>> 5 < 6
True
>>> 5 < 5
False
>>> 5 <= 5
True
>>> x = 1000
>>> y = 89
>>> x > y
True
>>> x == y
False
>>> 1000 == x
True
>>> s = "banana"
>>> s == "banana"
True
>>> True
True
>>> False
False
```

New: comparison operators

`< <= > >=`

Compare numbers, strings and more. Returns True or False depending on the relationship of the values

`==`

Works on any two values, returns True if equal

```
>>> True and False
False
>>> True or False
True
>>> True and True
True
>>> False or False
False
>>> False and False
False
>>> False and True


>>> False or True


>>> True or True
```

`and or`

Take two **booleans** and produces a new boolean:

`b1 and b2` is True when: ______________________________

`b1 or b2` is True when: ______________________________

```
= RESTART ...
>>> is_positive(9)
True
>>> is_positive(-1)
False
```

```
def is_positive(n):
    return n > 0


# write is_longer_than, which takes a string s and
# a number n and returns True if the string has
# more than n characters, and False otherwise.




# write a function between, which takes three
# numbers x, y, z and checks if y is between x and z
```

```
def my_abs(n):
  if n < 0:
    return n * -1
  else:
    return n




def letter_grade1(points):
  if points >= 90: return "A"
  elif points >= 80: return "B"
  elif points >= 70: return "C"
  else: return "F"

def letter_grade2(points):
  if points >= 90: return "A"
  elif points >= 80: return "B"
  elif points >= 70: return "C"
  elif points < 70: return "F"

def letter_grade3(points):
  if points < 70: return "F"
  elif points < 80: return "C"
  elif points < 90: return "B"
  else: return "A"

def letter_grade4(points):
  if points < 70: return "F"
  elif points >= 70: return "C"
  elif points >= 80: return "B"
  elif points >= 90: return "A"
```