```
# VARIABLES

# Creating a variable and assigning values
name = "Lucie"  # string type
number = 9  # integer type
decimal = 10.5 # float type (decimals)
silly = True # boolean type (True or False)

# Printing information from the variables
print(name)
print(number)
print(decimal)
print(silly)

# Using the data from variables
sum = number + decimal  # sum has the value 19.5
text = "Hello " + name + "!"  # text is "Hello Lucie!"

# Updating a variable with a NEW value
name = "Radeha"  # name was "Lucie" now it's "Radeha"
number = 99  # number was 9 now it's 99
decimal= 5.5  #decimal was 10.5 and now it's 5.5
silly = False
```

```
# BASIC MATH OPERATORS

add = 10 + 2    # add = 12
sub = 10 - 2    # sub = 8
mult = 10 * 2   # mult = 20
div = 10 / 2    # div = 5
mod = 10 % 3    # mod = 4
exp = 10 ** 2   # exp = 100
```

```
# STRING CONCATENATION

name = "Thomas"
age = 18

# Concatenating strings with strings
sentence1 = "Hello " + name + "!"

# Concatenating strings and numbers
sentence2 = name + " is " + str(age) + " years old."
# use the str(num) function to format a number as a string
# CONDITIONALS

# Comparison Operators
# value1 == value2   "equal to"
# value1 > value2    "greater than"
# value1 < value2    "less than"
# value1 <= value2   "less than or equal to"
# value1 >= value2   "greater than or equal to"
# value1 != value2   "not equal to"

               ###

# if
likesAnime = True

if likesAnime == True:
    print("I like anime too!")

# "I like anime too!" will only be printed if likesAnime has a value of
# True.  Nothing will be printed if likesAnime is False.

               ###

# if-else
pet = "cat"

if pet == "cat":
    print("meow")
else:
    print("woof")

#  "meow" is printed if pet has a value of "cat".
#  "woof" will be printed if pet does not have a value of "cat".

               ###

# if-elif-else
secret_number = 8
user_input = input("Guess my secret number: ")
guess = int(user_input)

if guess > secret_number:
    print("The secret number is less than " + str(guess))
elif guess < secret_number:
    print("The secret number is greater than " + str(guess))
else:
    print("You guessed my secret number!")

# Note: You can have more than one elif
```

```python
# FUNCTION EXAMPLES

# Function Example 1

# DEFINING the function
def greet(name):
    return "Hello " + name + "!"
# name is a "parameter"
# It's a variable that will recieve the input to the function.

# CALLING the function
print(greet("Alex"))
# The input to a function is also called an argument.
# "Alex" is sent to the greet function as input.
# The output of this function call is "Hello Alex!"

               ###

# Function Example 2
# DEFINING the function
def calculate_tip(bill, percent):
    tip = bill * percent/100
    return tip

# CALLING the function
get_bill = input("How much did your meal cost? ")
get_percent = input("What percent would you like to tip? ")
bill = float(get_bill)
percent = float(get_percent)
tip = calculate_tip(bill, percent)
print("You should tip your server $" + str(tip))
# The inputs to the calculate_tip function are bill and percent.
# The output of the function is the tip.

               ###
# Function Example 3
# DEFINING the function
def name_game(name):
    print(name + "!")
    print(name + ", " + name + " bo-" + name)
    print("banana fana fo-" + name)
    print("fee fi mo-" + name)
    print(name + "!")
# This function is different from the others because it does not have
# a return value.  The output is a printed chorus of the name game.

# CALLING the function
name_game("Maria")
# Calling name_game with an input(argument) of "Maria" results in an output of:
# Maria!
# Maria, Maria bo-Maria
# banana fana fo-Maria
# fee fi mo-Maria
# Maria!
# LISTS CRUD
# Create - How to create a new item in the data structure.
# Read - How to access an item in a data structure.
# Update - How to change an item in a data structure.
# Delete - How to remove an item from a list.

adventure_time_characters = ['Princess Bubblegum', 'Marceline', 'Lumpy Space Princess']

## CREATE ##
adventure_time_characters.append("Flame Princess")
# This adds the element "Flame Princess" at the end of the list.
# The list is now: ['Princess Bubblegum', 'Marceline', 'Lumpy Space Princess', 'Flame Princess']

adventure_time_characters.insert(2,"Tree Trunks")
# This inserts "Tree Trunks" at index 2.
# All of the elements shift to accordingly.
# The list is now: ['Princess Bubblegum', 'Marceline', 'Tree Trunks', 'Lumpy Space Princess', 'Flame Princess']

adventure_time_characters.extend(["Finn", "Jake", "Lemongrab"])
# This extends the list with Finn, Jake, and Lemongrab
# The list is now: ['Princess Bubblegum', 'Marceline', 'Lumpy Space Princess', 'Flame Princess', 'Finn', 'Jake', 'Lemongrab']

## READ ##
print(adventure_time_characters[1])
#This prints "Marceline" since that is the element at index 1.

print(adventure_time_characters[2])
#This prints "Lumpy Space Princess" which is the element at index 2.

print("Printing the element at the end of the list")
print(adventure_time_characters[len(adventure_time_characters)-1])
# Remember, Python starts counting at zero.  The first element in the list has
# an index of zero.  The second element has an index of one etc...
# The LAST ELEMENT will have an index of the length - 1.  So if a list has 5
# items, the last index will be 4.

## UPDATE ##
adventure_time_characters[2] = "BMO"
# This changes the element at index 2, from "Lumpy Space Princess" to "BMO"
# The list is now: ['Princess Bubblegum', 'Marceline', 'BMO', 'Flame Princess', 'Finn', 'Jake', 'Lemongrab']

## DELETE ##
adventure_time_characters.remove('Lemongrab')
# This removes "Lemongrab" from the adventure_time_characters list.
# The list is now:
# ['Princess Bubblegum', 'Marceline', 'BMO', 'Flame Princess', 'Finn', 'Jake']

adventure_time_characters.pop(4)
# This removes the element at index 4 ('Finn')
# The list is now:
# ['Princess Bubblegum', 'Marceline', 'BMO', 'Flame Princess', 'Jake']
# DICTIONARY CRUD
# Create - How to create a new item in the data structure.
# Read - How to access an item in a data structure.
# Update - How to change an item in a data structure.
# Delete - How to remove an item from a list.

#####################################################
# Pikachu's name, Pokedex, height, weight, type
# Source: https://www.pokemon.com/us/pokedex/pikachu


pika_dict = {
    "name": "Pikachu",
    "pokedex index": 25,
    "height": "16 inches",
    "weight": "13.2 lbs",
    "type": "electric"
}

# Dictionaries have "key":"value" pairs
# The "name" key has a value of "Pikachu"
# The "pokedex index" has a value of 25
# Note: Dictionaries are created with "mustache" brackets

print(pika_dict)

### CREATE ###
pika_dict["category"] = "mouse"
pika_dict["ability"] = "static"

# Creating a new key:value pair in the dictionary using the following syntax:
# dict_name[key] = value

### READ ###
print(pika_dict["name"])
print(pika_dict["pokedex index"])

# Read information from the dictionary using this syntax:
# dict_name[key]

### UPDATE ###
pika_dict["name"] = "Pika"
print(pika_dict["name"])

pika_dict["height"] = "17 inches"
print(pika_dict["height"])

# Upadte values associated with a key using the "=" sign.
# = is the assignment operator

### DELETE ###
del pika_dict["weight"]

# To delete key-value pairs use the del command.
# Syntax: del dict_name[key]
```

```
### ITERATING THROUGH A DICTIONARY ###
for item in pika_dict:
    print(item)

# This prints all of the keys in the dictionary

for key in pika_dict:
    print(pika_dict[key])

#This prints all of the values in the dictionary
```