

The background is a solid purple color. On the left side, there are several overlapping circles of different shades of purple. Scattered around these circles are several small yellow dots of varying sizes.

Exercises 1 - 5

Pair Programming

- an agile software development technique
- two programmers work together at one workstation
- one, the driver, writes code
- the other (observer or navigator), dictates & reviews each line of code as it is typed in
- the two programmers switch roles frequently

1. Create & Run Python file from command-line
2. Comments & Pound Characters
3. Numbers and Math
4. Variables & Names
5. More Variables & Printing

1. Create & Run Python file from CLI

create file:

```
1 touch ex01.py
```

run file:

```
1 python3 ex01.py
```


ex01.py

```
1 print("Hello World")
2 print("Hello Again")
3 print()
4 print("Hello BootCamper's Collective!")
5 print("My name is (your name here).")
6 print()
7 print("'print()' is a python method.")
8 print("Python Docs are here --> https://docs.python.org/3/")
9 print()
10 print("This line has nested 'single' quotes.")
11 print('This line has nested "double quotes".')
12 print()
13 print('Now toggle into your terminal and type: python3 ex1.py')
14 print('This symbol --> # <-- is called an "octothorpe", a "mesh", "pound", or "hash".')
15
16 # print("Remove the hash from the beginning of this line, so the python interpreter can print this text.")
```

(on completion, run from command line)

2. Comments & Pound Characters

ex02.py

```
1 # A comment, this is so you can read your program later.
2 # Anything after the # is ignored by python.
3
4 print("You can have code like this.") # and the comment after is ignored.
5
6 # You can also use a comment to "disable" or comment out code:
7 # print("This won't run.")
8
9 print("This will run.")
10
11 # Now run ex02.py...--> python3 ex02.py <-- just what's in the arrows :-)
```


3. Numbers & Math (1 of 2)

ex03.py

```
1 # +: plus
2 # -: minus
3 # /: slash
4 # *: asterisk
5 # %: percent
6 # <: less-than
7 # >: greater-than
8 # <=: less-than-equal
9 # >=: greater-than-equal
10
11 print("I will now count my chickens:")
12
13 print("Hens", 25 + 30 / 6)
14 print("Roosters", 100 - 25 * 3 % 4)
15
16 print("Now I will count the eggs:")
```


3. Numbers & Math (2 of 2)

ex03.py

line 18

```
print(3 + 2 + 1 - 5 + 4 % 2 - 1 / 4 + 6)
```

```
print("Is it true that 3 + 2 < 5 - 7?")
```

```
print(3 + 2 < 5 - 7)
```

```
print("What is 3 + 2?", 3 + 2)
```

```
print("What is 5 - 7?", 5 - 7)
```

```
print("Oh, that's why it's false.")
```

```
print("How about some more.")
```

```
print("Is it greater?", 5 > -2)
```

```
print("Is it greater or equal?", 5 >= -2)
```

```
print("Is it less or equal?", 5 <= -2)
```


4. Variables & Names (1 of 2)

ex04.py

```
1 # Comment above each line, explaining what is happening to the line below
2
3 # The integer 100 is assigned to the variable 'car'
4 cars = 100
5
6 # The space available for a passenger in each car
7 space_in_car = 4.0
8
9 # The number of people available to drive a car
10 drivers = 30
11
12 # The number of non-drivers who need available space in a car
13 passengers = 90
14
15 # The number of cars not driven is equal to the number of cars in the fleet
16 # minus the number of available drivers
```


ex04.py (2 of 2)

continue on line 17



```
cars_not_driven = cars - drivers

# A car is 'driven' when a 'driver' operates it
cars_driven = drivers

# The total amount of available space in the entire fleet of cars
carpool_capacity = cars_driven * space_in_car

# Because each car needs a driver, dividing that into the number of passengers
# gives us the average number of passengers per car
average_passengers_per_car = passengers / cars_driven

print("There are", cars, "cars available.")
print("There are only", drivers, "drivers available.")
print("There will be", cars_not_driven, "empty cars today.")
print("We can transport", carpool_capacity, "people today.")
```

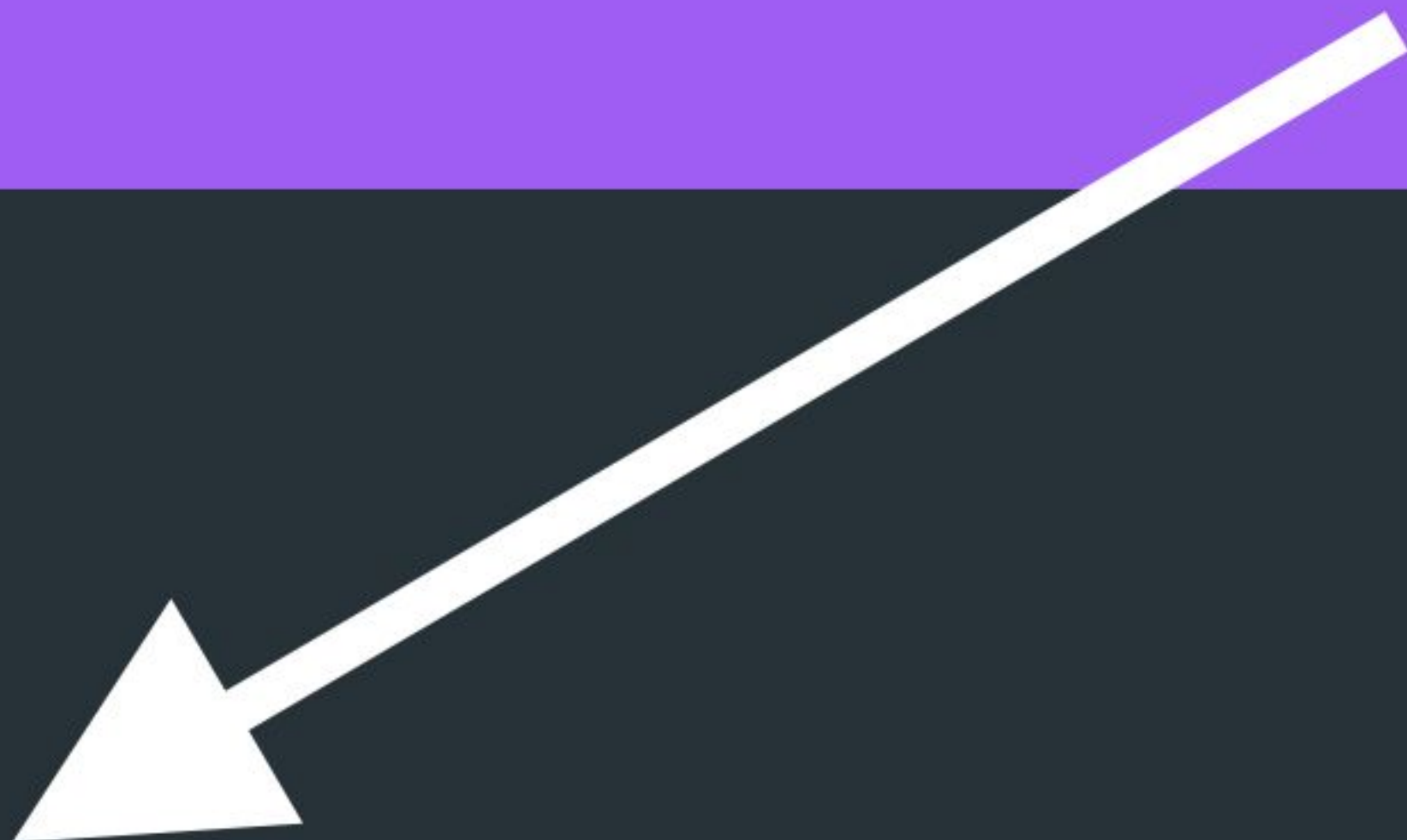

5. More Variables & Printing (1 of 2)

ex05.py

```
1 # We will make strings with variables imbedded in them.
2 # Supply your own data.. :-)
3
4 my_name = "first last" # for example: my_name = "Andy Young"
5 my_age = 38
6 my_height = 66 # inches
7 my_weight = 160 # pounds
8 my_eyes = "green"
9 my_teeth = "white"
10 my_hair = "brown"
11
12 # The 'f' before the string tells python to 'format' the string.
13 print(f"Let's talk about {my_name}.")
14 print(f"He's {my_height} inches tall.")
15 print(f"He's {my_weight} pounds heavy.")
16 print("Actually that's not too heavy.")
```


5. More Variables & Printing (2 of 2)

ex05.py



```
print("Actually that's not too heavy.")
print(f"He's got {my_eyes} eyes and {my_hair} hair.")
print(f"His teeth are usually {my_teeth} depending on the coffee.")

# this line is tricky, try to get it exactly right
total = my_age + my_height + my_weight
print(f"If I add {my_age}, {my_height}, and {my_weight} I get {total}.")
```