Which of these list comprehensions will create a list equal to `desired_list`?

```
desired_list = [-1, 0, 1, 2, 3]
```

```
[i-1 for i in range(5)]
```

👏 Correct! `range(5)` will contain the numbers 0 through 4, and the list comprehension will contain each of those minus one.

```
range(5)
```

```
desired_list = []
for i in range(5)
    desired_list = i - 1
```

What would be the output of the following code:

```
numbers = [1, 1, 2, 3]
for number in numbers:
    if number % 2 == 0:
        break
    print(number)
```

```
1
1
3
```

```
2
```

```
1
1
2
3
```

```
1
1
```

What would be the output of the following code:

```python
drink_choices = ["coffee", "tea", "water", "juice", "soda"]
for drink in drink_choices:
    print(drink)
```

coffee

drink

["coffee", "tea", "water", "juice", "soda"]

coffee
tea
water
juice
soda

👏 Correct! The loop prints each element of the list!

Fill in the code to loop over the list `grouped_topics` and print every element in the list.

```python
grouped_topics = [["Algorithms", "Data Structures", "AI"], ["Linear Regression", "SQL"]]
```

```python
for sublist in   grouped_topics   :

    for sublist_element in   sublist   :

        print(   sublist_element   )
```

👏 You got it!

Fill in the blank with the appropriate `while` condition in order to print the numbers 1 through 10 in order:

```
i = 1
___Blank___
    print(i)
    i += 1
```

```
while i > 10:
```

```
while i <= 10:
```

👏 Correct! This will print the counter and then increase the counter by one, until the counter is `11` and the loop body is not entered.

```
while i < 10:
```

```
while i == range(11):
```

What would be the output of the following code:

```
for i in range(3):
    print(i)
```

i
i
i

0
1
2

👏 Correct! The `range()` function starts at 0.

Which of these list comprehensions will create a list equal to `desired_list`?

```python
my_list = [5, 10, -2, 8, 20]
desired_list = [10, 8, 20]
```

```python
[i for i in my_list if i > 5]
```

👏 Correct! This will take the elements of `my_list` that are greater than 5, which are `10`, `8`, and `20`.

```python
[i for i in my_list]
```

```python
[i for i in my_list if i > 10]
```

What would be the output of the following code:

```python
numbers = [1, 1, 2, 3]
for number in numbers:
    if number % 2 == 0:
        continue
    print(number)
```

```
1
1
3
```

👏 Correct! When the loop reaches the element `2`, it passes to the next number before printing it.

What would be the output of the following code:

```
for i in range(3):
    print(5)
```

3
3
3
3
3

5
5
5

👏 Correct! The code inside the loop (printing 5) is executed 3 times.

What would be the output of the following code:

```
numbers = [2, 4, 6, 8]
for number in numbers:
    print("hello!")
```

hello!

2 hello! 4 hello! 6 hello! 8 hello!

2 4 6 8

hello!
hello!
hello!
hello!

👏 Correct! The loop prints "hello!" for each of the 4 numbers in numbers.