Python Functions Exercise

1: Create a function that can accept two arguments name and age and print its value

2: Write a function `func1()` such that it can accept a variable length of argument and print all arguments value

```
func1(20, 40, 60)
func1(80, 100)
```

Expected Output:

```
func1(20, 40, 60)

20

40

60



func1(80, 100)

80

100
```

3: Write a function calculation() such that it can accept two variables and calculate the addition and subtraction of them. And also it must return both addition and subtraction in a single return call

Given:

```
def calculation(a, b):
    # Your Code

res = calculation(40, 10)
print(res)
```

Expected Output

```
50, 30
```

4: Create a function showEmployee() in such a way that it should accept employee name, and its salary and display both. If the salary is missing in the function call assign default value 9000 to salary

Given:

```
showEmployee("Ben", 9000)
showEmployee("Ben")
```

Expected output:

```
Employee Ben salary is: 9000
Employee Ben salary is: 9000
```

5: Create an inner function to calculate the addition in the following way

- Create an outer function that will accept two parameters, `a` and `b`
- Create an inner function inside an outer function that will calculate the addition of `a` and `b`
- At last, an outer function will add 5 into addition and return it

6: Write a recursive function to calculate the sum of numbers from 0 to 10

Expected Output:

55

7: Assign a different name to function and call it through the new name

Below is the function `displayStudent(name, age)`. Assign a new name `showStudent(name, age)` to it and call through the new name

```
def displayStudent(name, age):
    print(name, age)

displayStudent("Emma", 26)
```

You should be able to call the same function using

```
showStudent(name, age)
```

8: Generate a Python list of all the even numbers between 4 to 30

Expected Output:

```
[4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28]
```

9: Return the largest item from the given list

```
aList = [4, 6, 8, 24, 12, 2]
```

Expected Output: