

Assignment 7 : Write a python to create user-defined functions with different types of function arguments

SOURCE CODE :

```
# Function with positional arguments
def positional_args(a, b):
    return a + b

# Function with keyword arguments
def keyword_args(a, b, c):
    return a * b + c

# Function with default arguments
def default_args(a, b=10):
    return a + b

# Function with variable-length arguments (*args)
def variable_length_args(*args):
    return sum(args)

# Function with variable-length keyword arguments (**kwargs)
def variable_length_kwargs(**kwargs):
    result = ""
    for key, value in kwargs.items():
        result += f"{key}: {value}\n"
    return result

def main():
    # Positional arguments
    print("Positional Arguments:")
    print(positional_args(5, 3)) # Output: 8
```

```
# Keyword arguments
```

```
print("\nKeyword Arguments:")
```

```
print(keyword_args(a=2, b=3, c=4)) # Output: 10
```

```
# Default arguments
```

```
print("\nDefault Arguments:")
```

```
print(default_args(5)) # Output: 15
```

```
print(default_args(5, 20)) # Output: 25
```

```
# Variable-length arguments
```

```
print("\nVariable-Length Arguments (*args):")
```

```
print(variable_length_args(1, 2, 3, 4, 5)) # Output: 15
```

```
# Variable-length keyword arguments
```

```
print("\nVariable-Length Keyword Arguments (**kwargs):")
```

```
print(variable_length_kwargs(name="Alice", age=30, city  
    ="Wonderland"))
```

```
if __name__ == "__main__":
```

```
    main()
```

OUTPUT :

Output

Positional Arguments:

8

Keyword Arguments:

10

Default Arguments:

15

25

Variable-Length Arguments (*args):

15

Variable-Length Keyword Arguments (**kwargs):

name: Alice

age: 30

city: Wonderland

=== Code Execution Successful ===