QUESTION 1

C

The code snippet is a recursive function that calculates the greatest common divisor (GCD) of two numbers using the Euclidean algorithm. The initial call is `func(30, 75)`. The function checks if `a` is equal to 0, and if so, it returns `b`. Otherwise, it makes a recursive call with the arguments `b % a` and `a`. The GCD of 30 and 75 is 15

QUESTION 2

B

The code defines a list of numbers. It then uses the `sorted()` function to sort the numbers in ascending order. The `filter()` function is used with a lambda function `even` to filter out only the even numbers from the sorted list. The `filter()` function returns a filter object, and the `type()` function is used to determine the type of the object.

QUESTION 3

A

The question asks about the data type in which `\*args` (variable-length arguments) are stored when passed into a function. In Python, when using `\*args` in a function definition, the arguments are stored as a tuple.

QUESTION 4

D

The code defines three sets: `set1`, `set2`, and `set3`. The `+` operator is used to concatenate the sets, but the `+` operator is not supported for sets in Python. It results in a TypeError.

QUESTION 5

A

The question asks for the keyword used in Python to raise exceptions. In Python, the `raise` keyword is used to raise exceptions.

QUESTION 6

C

The question does not provide a code snippet, but it asks for the correct module name to handle date and time computations in Python. Datetime is the correct module for date and time computations in Python.

QUESTION 7

C

The code snippet performs mathematical operations. It calculates `4\*\*3`, which is 64, and `(7 + 5)\*(1 + 1)`, which is 144. Then, it adds the results together, resulting in 208.

QUESTION 8

C

The question asks for the function that converts a date to the corresponding time in Python. Both a) `strptime()` and b) `strftime()`. The `strptime()` function is used to parse a string representing a date and time, and the `strftime()` function is used to format a date and time as a string.

QUESTION 9

B

The question asks about the nature of Python tuples. Tuples in Python are immutable, meaning their values cannot be changed after creation.

QUESTION 10

A

The question asks for the built-in function in Python that returns a range object for iterating over a series of integer numbers.

QUESTION 11

C

The question asks for the function that does not have any name. Lambda functions are anonymous functions without a name.

QUESTION 12

C

The question asks about the purpose of the Pickle module in Python. Both A and B. The Pickle module is used for serializing (pickling) and deserializing (unpickling) Python object structures.

QUESTION 13

B

The question asks about the method used to convert Python objects for writing data in a binary file. The `dump()` method is used in conjunction with the `pickle` module to serialize and write Python objects to a binary file

QUESTION 14

A

The question asks about the method used to unpickle data from a binary file. The `load()` method is used in conjunction with the `pickle` module to read and deserialize Python objects from a binary file.

QUESTION 15

D

The question asks what a text file contains. All of the mentioned above. A text file can contain alphabets, numbers, and special symbols depending on the content of the file.

QUESTION 16

A

The question asks which Python code could replace the ellipsis (...) to get a specific output. The desired output is "Enterprise Picard, Voyager Janeway, Defiant Sisko". items(): print(ship, captain)` will produce the desired output by iterating over the `captains` dictionary and printing the ship and captain names.

QUESTION 17

D

The question asks which line of code will create an empty dictionary named `captains`. `captains = {}` will create an empty dictionary named `captains`.

QUESTION 18

B

The question asks how to add key-value pairs to an existing dictionary named `captains`. `captains["Enterprise"] = "Picard" captains["Voyager"] = "Janeway" captains["Defiant"] = "Sisko"` will successfully add the key-value pairs to the existing `captains` dictionary.

QUESTION 19

B

The question asks how to display the ship and captain names from the dictionary `captains` with additional context. Among the given options, `for ship, captain in captains.items(): print(f"The {ship} is captained by {captain}.")` will display the ship and captain names with the desired context.

QUESTION 20

C

The question asks how to remove the entry for the key "Discovery" from the dictionary `captains`. `del captains["Discovery"]` will remove the entry for the key "Discovery" from the dictionary.