Python Assessment:

1) Difference between Django, Pyramid and Flask:

Ans:

1. **Django** comes with multiple stuff right of the box to make developer easier but your application is quite heavy for your server and thus bit slower. It offers a lot of out-of-the-box functionalities like Admin Panel or Generic Views and Forms. Provide MTV (model-template-view) architecture pattern. Django is a good fit for bigger projects.
2. **Flask** It’s one of the most popular Python microframeworks, it’s reliable and fast. it doesn’t offer ORM, but SQLAlchemy or other can be used. This framework will do the trick in small and medium projects.
3. **Pyramid** Pyramid starts with a minimal installation that can be extended when needed. It’s worth noting that it’s a part of the Pylons project, which integrates web-related technologies. there is no specific ORM but SQLAlchemy is recommended, This framework might be a good choice in cases when you don’t want to spend time learning custom frameworks solutions (like ORM) but still need an extensive tool for building software, as Pyramid supports usage of many well-known stand-alone solutions.

2) if a list is nums = [0,1,2,3,4] what is nums[-1]

Ans: nums[-1] = 4

3) Explain the output of the following piece of code

tuple=(123,John)

tuple\*=2

tuple

Ans: (123, John, 123, John)

4) Difference between append and extend in list:

Ans: append - Adds its argument as a single element to the end of a list. The length of the list increases by one. If you append another list onto a list, the parameter list will be a single object at the end of the list.

e.g

my\_list = ['I', 'Like', 'Cricket']

another\_list = [6, 0, 4, 1]

my\_list.append(another\_list)

print my\_list

output = ['I', 'Like', 'Cricket', [6, 0, 4, 1]]

extend: Iterates over its argument and adding each element to the list and extending the list. The length of the list increases by number of elements in it’s argument.

e.g

my\_list = ['I', 'Like', 'Cricket']

another\_list = [6, 0, 4, 1]

my\_list.append(another\_list)

print my\_list

output = ['I', 'Like', 'Cricket',6, 0, 4, 1]

5) How do you remove the leading whitespace in a string? For example, leading whitespace in a string is the whitespace in a string before the first non-whitespace character. Eg. ' Maersk'

Ans: We can remove leading whitespace in string using lstrip method in python.

e.g

text = ' Maersk'

text = text.lstrip()

print(text)

output: Maersk

6) What is the enumerate () function in Python? Explain with an example.

Ans: It is built-in function. The enumerate functions allows you to loop over an iterable object and keep track of how many iterations have occurred Enumerate is particularly useful if you have an array of values that you want to run through entirely. It takes the given input as a collection or tuples and returns it as an enumerate object.

e.g mylist = ["I","am","happy"]

print(list(enumerate(mylist)))

output: [(0, 'I'), (1, 'am'), (2, 'happy')]

7) Explain atleast three advantages of NumPy Array over the list in python.

Ans: Advantages numpy array over python list:

1) Numpy array is fast computing compare to list for nested list for manipulating every element of multidimensional data

2) Numpy array consumes less memory

3) convenient to use

4) Numpy provides a mechanism of specifying the data types.

8) List out all the possible differences between method and constructor in Python.

Ans:

Method: Method has to be invoked. method has return type. method work for initiated object.

Constructor: Constructor automatically gets invoked. Constructor has no return type. Constructor initiates the object.

9) Define generator and iterator with an example in Python. What is Monkey Patching?

Ans: generator: Python provides a generator to create your own iterator function. A generator is a special type of function which does not return a single value, instead, it returns an iterator object with a sequence of values. In a generator function, a yield statement is used rather than a return statement.

e.g

def mygenerator():

print('First item')

yield 10

Iterator: An iterator is an object which contains a countable number of values and it is used to iterate over iterable objects like list, tuples, sets, etc. Iterators are implemented using a class and a local variable for iterating is not required here, It follows lazy evaluation where the evaluation of the expression will be on hold and stored in the memory until the item is called specifically which helps us to avoid repeated evaluation.

e.g

>>> gen = mygenerator()

>>> next(gen)

First item

10

Moneky Patching: the term monkey patch only refers to dynamic modifications of a class or module at runtime, motivated by the intent to patch existing third-party code as a workaround to a bug or feature which does not act as desired.

10) What will the output of the following code snippet:

def squares(n):

i=1

while(i<=n):

yield i\*\*2

i+=1

for i in squares(7):

print(i)

Ans:

Output: 1

4

9

16

25

36

49

**Embedded Theory Questions**

**1) H**ow I/O devices are classified for embedded system?

**Ans:** Input/output devices are usually called I/O devices.They are directly connected to an electronic module inside the systems unit called a device controller. For example, the speakers of a multimedia computer system are directly connected to a device controller called an audio card (such as a Soundblaster), which in turn is connected to the rest of the system.Sometimes secondary memory devices like the hard disk are called I/O devices (because they move data in and out of main memory) .What counts as an I/O device depends on context. To a user, an I/O device is something outside of the system box.

2) What is the difference between Microprocessor and Microcontroller?

Ans: Ultimately, microcontrollers and microprocessors are different ways of organizing and optimizing a computing system based on a CPU. While a microcontroller puts the CPU and all peripherals onto the same chip, a microprocessor houses a more powerful CPU on a single chip that connects to external peripherals.

3) What is a Watchdog Timer?

Ans: A watchdog timer is an electronic or software timer that is used to detect and recover from computer malfunctions. Watchdog timers are widely used in computers to facilitate automatic correction of temporary hardware faults, and to prevent errant or malevolent software from disrupting system operation.

4) What are common errors in Embedded system?

Ans: The standard errors that are found in an embedded system are: Stack overflow error :- Call stack has a limited amount of memory, and excessive memory usage in this stack can cause the failure. The memory of the call stack contains information of active subroutines in the program.

5) What is the need for an infinite loop sometimes in embedded systems?

Ans: The infinite loop is necessary because the embedded software's job is never done. It is intended to be run until either the world comes to an end or the board is reset, whichever happens first. In addition, most embedded systems have just one piece of software running on them.