**ASSIGNMENT OF PYTHON SESSION # 1**.

**Q. 1. What is JPython & CPython ?.**

Ans. **CPython –**

CPython is a version of the Python Interpreter that is written in the C Programming Language. CPython is the original and usually default interpreter used to run Python programs, and is the interpreter currently developed by Python **creator Guido Van Rossum**. Currently the CPython Interpreter supports versions 2.7 and 3.4.

**Jython or JPython –**

Jython is the JVM implementation of the Python programming language. It is designed to run on the Java platform. A Jython program can import and use any Java class. Just as Java, Jython program compiles to **bytecode**. One of the main advantages is that a user interface designed in Python can use GUI elements of **AWT**, **Swing** or **SWT Package**.

**Q. 2. Difference between ASCII format and Unicode format.**

Ans. **ASCII –**

**American Standard Code for Information Interchange (ASCII)** is a character-encoding scheme and it was the first character encoding standard. It is a code for representing English characters as numbers, with each letter assigned a number from 0 to 127. Most modern character-encoding schemes are based on ASCII, though they support many additional characters.

**Unicode –**

**Unicode** is a standard which defines the internal text coding system in almost all operating systems used in computers at present. Unicode assigns each character a unique number, or code point. It defines two mapping methods, the UTF (Unicode Transformation Format) encodings, and the UCS (Universal Character Set) encodings. Unicode-based encodings implement the Unicode standard and include UTF-8, UTF-16 and UTF-32/UCS-4.. The first 128 characters are the same as the ASCII system making it compatable. There are 6400 characters set aside for the user or software.

**Q. 3. Basic difference between Python2 & python3.**

Ans.

* Print: In Python 2, “print” is treated as a statement and Python 3 explicitly treats “print” as a function, which means you have to pass the items you need to print to the function in parentheses in the standard way, or you will get a syntax error. a statement rather than a function.
* Integer Division: Python 2 treats numbers that you type without any digits after the decimal point as integers, which can lead to some unexpected results during division. For example, if you type the expression 3 / 2 in Python 2 code, the result of the evaluation will be 1, not 1.5 as you might expect.
* List Comprehension Loop Variables: In previous versions of Python, giving the variable that is iterated over in a list comprehension the same name as a global variable could lead to the value of the global variable being changed — something you usually don’t want. This irritating bug has been fixed in Python 3, so you can use a variable name you already used for the control variable in your list.
* Unicode Strings: Python 3 stores strings as Unicode by default, whereas Python 2 requires you to mark a string with a “u” if you want to store it as Unicode. Unicode strings are more versatile than ASCII strings, which are the Python 2 default, as they can store letters from foreign languages as well as emoji and the standard Roman letters and numerals.
* Raising Exceptions: Python 3 requires different syntax for raising exceptions. If you want to output an error message to the user, you need to use the syntax:

.