CST2101 Business Intelligence Programming

Lab 1: Programming Languages and Objects

DUE	MATERIALS	OBJECTIVE	EVALUATION
06 October 2017	Student Laptop	Demonstrate an understanding of programming languages and objects	marks maximum 5

Student Name: Carlos Timoteo   Student Number: 040905599

STEP	INSTRUCTIONS
I	**Design an Object Model**

1
Mark
Spend some time in the kitchen this week and contemplate the oven. If you were asked to build a software model (i.e.create an object) of an oven, identify 5 properties that your oven has. For each property, identify a reasonable range of values and identify the Python data type that best suits that property. (0.1 mark for each property and 0.1 marks for each correct Data Type)

PROPERTY	Python DATA TYPE
a) Elements_state	Elements_state [ElementState enum]. Ex: class ElementState(Enum): ON=1 OFF=0 Elements_state=[] : ElementState List of Enum
b) Oven_temperature	Oven_temperature : int
c) Oven_mode	Class OvenMode(Enum): Broil = 0 Bake = 1
d) Elements_temperature	Elements_temperature=[]:int List of integer
e) OvenTimer	Class OvenTimer(value) : Def notifyCook() timer = Timer(value, notifyCook)

Lab 1: Programming Languages and Objects

II	**High Level Languages**

1
Mark

Research each of the following high-level languages and identify the types of problems or industry area where the language is mostly used. (0.2 marks each)

LANGUAGE	DOMAIN or PROBLEM AREA
JavaScript	JavaScript is a lightweight, interpreted programming language with object-oriented capabilities. In the past, its common usage was running dynamic web pages in client web browser. In other words, it allows you to create highly responsive interfaces that improve the user experience and provide dynamic functionality. Nowadays, JavaScript can be used to build server-side applications using modern frameworks like Node.js. Besides that, Angular and others frameworks are being used to decorate HTML pages providing easy development of dynamic, interactive, reactive and server-less (adopting RESTful architecture) web applications. (Flanagan, 2006) (Jain, 2015) (Cantelon, 2014)
JOVIAL	JOVIAL is a high-level computer programming language specialized for the development of embedded systems. Jovial includes features such as assembly level inserts, records and array of records, and is used extensively in USAF applications. Jovial was used extensively in the 1970s and 1980s to develop software for a broad range of military and aerospace systems. Among these are the B-52, B-1, and B-2 bombers, C-130, C-141, and C-17 transport aircraft, F-15, F-16, F-18, and F-117 fighter aircraft, E-3 Sentry AWACS aircraft, Navy Aegis cruisers, Army Multiple Launch Rocket System (MLRS), and the Army UH-60 Black Hawk helicopters. (JOVIAL - Wikipedia, 2017)
COBOL	Computational and mathematics intensive applications, used mostly by business, finance, and administrative systems for companies and governments. COBOL is still widely used in legacy applications deployed on mainframe computers, such as large-scale batch and transaction processing jobs.

Lab 1: Programming Languages and Objects

	COBOL's numeric processing functions make it a good choice for applications where the tiniest fractional rounding error can make a crucial difference. COBOL systems process data quickly. It is fast. It integrates with every system architecture. It is imperative, procedural and, since 2002, object-oriented. (Wikipedia - COBOL, 2017)
Objective-C	Objective-C is a general-purpose, object-oriented programming language that adds Smalltalk-style messaging to the C programming language. It was the main programming language used by Apple for the OS X and iOS operating systems, and their respective application programming interfaces (APIs) Cocoa and Cocoa Touch prior to the introduction of Swift. Objective-C is a fundamentally simple language. Its syntax is small, unambiguous, and easy to learn. Object-oriented programming, with its self-conscious terminology and emphasis on abstract design, often presents a steep learning curve to new recruits. A well-organized language like Objective-C can make becoming a proficient object-oriented programmer that much less difficult. Compared to other object-oriented languages based on C, Objective-C is very dynamic. It supports an open style of dynamic binding, a style that can accommodate a simple architecture for interactive user interfaces. Dynamism enables the construction of sophisticated development tools. (Why Objective-C, 2017) (Wikipedia - Objective-C, 2017)
MAPLE	Maple is a symbolic and numeric computing environment, and is also a multi-paradigm programming language, designed for mathematics. Maple engine is used within several other products from Maplesoft: MAPLE T.A., MapleNet, MapleSim, Mathcad, MATLAB. (Wikipedia - MAPLE, 2017)

Lab 1: Programming Languages and Objects

III	Compare Python

3 Marks Choose either Java, JavaScript or C++ and research the key differences between your chosen language and Python. You must identify at least six differences and note your sources (only the webpage or book/page number are required).

You may use bullet form as long as your comparisons are clear. Your answer for this section cannot exceed one page.

Python (Toal, 2016):
- Uses indentation to define structure;
- Is known for being easy to read and write;
- Has a large number of data types, including tuples, lists, sets, and dictionaries are built-in with convenient syntactic forms. The language comes with a huge standard library;
- Is particularly powerful as a scripting language, but it is a truly general-purpose language;
- Used in domains from games and entertainment to web applications to statistics and data science;
- Has a rich type system, with types organized into an ontology that supports something called multiple inheritance;
- Has conditional expression in the form y if x else z;
- Has an unlimited number of types;
- A type is a object. Everything is type of object;
- Has only object, no primitives;
- Python types have supertypes and subtypes;
- Python variables are either global or scoped to a function. There is no explicit variable declaration; assigning to a variable inside a function creates a local variable, unless marked global or nonlocal;
- Passes arguments to parameters by passing each argument's id, this mechanism is known as pass-by-sharing;
- Supports both default arguments and argument packing and unpacking. A single-starred parameter packs any additional arguments into a tuple. In a call, the single-star unpacks a sequence so that its constituent elements are passed to multiple parameters;
- Calls arguments passed in this fashion keyword arguments;
- Features classical inheritance, in other words, an object's attributes come from its class and all of the class's ancestors;
- In Python, functions (and hence methods) can be decorated, allowing us to do some extra work when the function is called;

Lab 1: Programming Languages and Objects

Javascript (Toal, 2016):
- The syntax was strongly influenced by C, with curly braces, assignment statements, and the ubiquitous if, while, and for statements;
- Functions are first-class values: they can be assigned to variables, passed to functions, and returned from functions;
- Yet the attempt to keep the language simple led to several notorious features as well. Weak typing, where expressions of the wrong type are automatically coerced to "something that works," and automatic semicolon insertion, where the language will figure out where your statements begin and end when you are not explicit, save typing but sometimes produce utterly surprising behavior;
- JavaScript's success is not limited to the browser; the language powers server-based applications supporting thousands of concurrent users;
- Has a small, fixed, number of types. It is only 7 types;
- In JavaScript asking for the type of the value 5 produces the string "number" in reply;
- Implements higher-order functions, i. e. functions that accept functions as parameters or return functions;
- Has primitives and objects;
- The class constructor is syntactic sugar—a syntax that makes the standard form easier to read. It does no more and no less than defining the function and assigning methods to the prototype;
- Requires the programmer to set up prototype chains to simulate IS-A relationships between types;
- Is statically scoped;
- Features prototypal inheritance.
- JavaScript's asynchronous *callback's* and *promises* comprise the only two ways to manage concurrency.

IV   Submit Your Work

- Submit your work on this sheet with you name and student number at the top or in a Microsoft Word document.

Submit all materials through Blackboard.

Lab 1: Programming Languages and Objects

References

Cantelon, M. H. (2014). *Node. js in Action.* Manning.

Flanagan, D. (2006). *JavaScript: the definitive guide.* O'Reilly Media, Inc.

Jain, N. M. (2015). AngularJS: A modern MVC framework in JavaScript. *Journal of Global Research in Computer Science*, 5(12), 17-23.

JOVIAL - Wikipedia. (2017, October 05). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/JOVIAL

Toal, R. R. (2016). *Programming Language Explorations.* CRC Press.

Why Objective-C. (2017, October 5). Retrieved from Apple Developer: https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/OOP_ObjC/Articles/ooWhy.html

Wikipedia - COBOL. (2017, October 5). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/COBOL

Wikipedia - MAPLE. (2017, October 5). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Maple_(software)

Wikipedia - Objective-C. (2017, October 5). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Objective-C