Assignment 2: Parallel Implementation

DOCUMENTATION

YOUR NAME

Contents

[Requirements: 3](#_Toc9887633)

[Class Diagram: 3](#_Toc9887634)

[Use Case Diagram: 5](#_Toc9887635)

[Sequence Diagram: 6](#_Toc9887636)

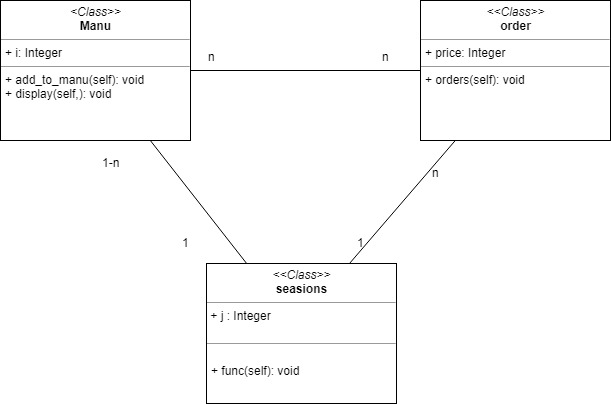
[Languages Comparison: 7](#_Toc9887637)

[Comparative overview of the languages: 9](#_Toc9887638)

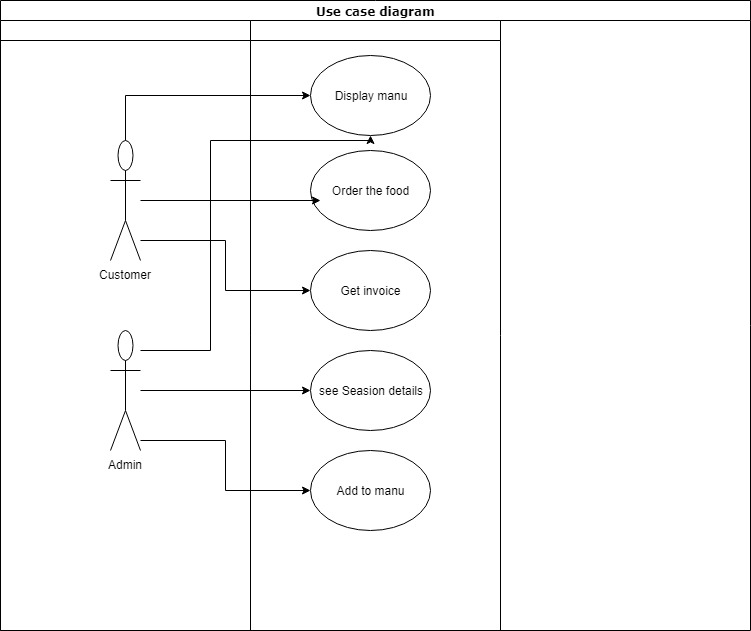
# Requirements:

* Calculate prices of the order items. i.e. pizza and pasta.
* Print total earnings form pizzas and pastas.
* Print total count of pizzas and pastas.

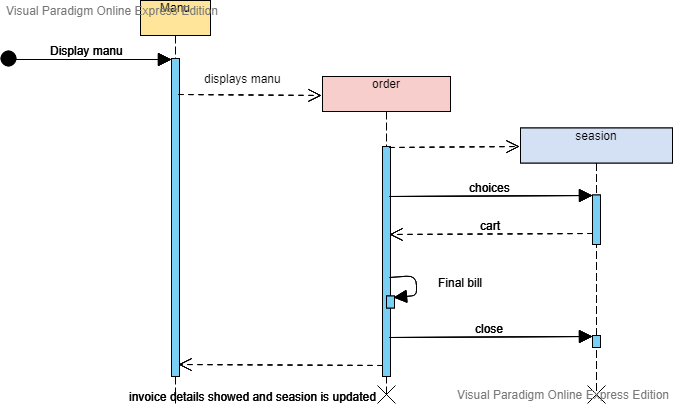
# Class Diagram:



# Use Case Diagram:



# Sequence Diagram:



# Languages Comparison:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Features Used | Advantages | Lacking Features |
| C | Arrays  Loops  Variables  Functions  Inputs  Output Statements  Conditional statements Break Statements | Extremely fast  Functional programming  Syntax is easy  C is highly portable  Data structure support is amazing  Has an easy Syntax | Pointers are very complex.  No support for object oriented programming.  Difficult for low level coding  Does not have classes so no data encapsulation |
| Java | Arrays  Loops  Classes  Print statement  Functions  Variables  Polymorphism  Input from console  Conditional statements Break Statements | Easy to debug  High support of Object oriented programing  Platform independent  A lot of support online  Easy to compile and run | Slow in execution  Takes more memory  Need a specific compiler and JDE  We cannot control garbage collector so it delete or collect the variable without our consent  There are no unassigned variables like in c |
| Lisp | Variables  Loops  Output statements of all type  If when statement  Prog to prolong the conditional statements  Functions  Unless for comparison  Defining and setting variables  Functions with parameters  Case statements  Conditional statements Taking user input | Functions can be passed as values in variables  Can be used to generate code  Simple and executes on CMD or Console  Lisp was ahead of the time when it was used | Complex syntax  Hard to code as every line in code starts and ends with round brackets  It is hectic to use different format to print different kind of variables  The syntax was different and somewhat obsolete  Does not supports low level programming like pointers and |
| Python | Lists  Inheritance  Functions  Loops  Conditional statements  Class  Return  Parameterized functions | High level language  Easy to run the code  Cross operating system support  Easy to code  Variables don’t have a type they are assigned types on the run time  Python is the new rising trend | Slow in execution  Need to take care of the indentation  Requires a good environment to code in  I used Jupiter notebook so mostly the errors were found after execution  Memory consumption is also an issue with python  It has issues with data base or we can say that other languages have a better database access |

# Comparative overview of the languages:

As far as the implementation is concerned, Java was the easiest to implement because of the nature of having closeness to the object oriented programming. Like the item is said to be object oriented and the pizza and pasta were derived from it. This means that if a new item needs to be added it should just only make a new object of the class and that’s it. The code environment has provided the facility of manually telling errors and suggesting to auto correct them which was very helpful. Python nearly followed the same pattern but there was some difficulty due to the structure of the coding in it. Lisp and C both didn’t support object oriented and the work done was to be done with the help of functions so that became quite complex. Also I considered the online support available for both of them and Java was most preferable as C and Lisp are old and have a difficult syntax so I prefer to use java because syntax is somewhat easy and it also supported the things that I wanted to implement As java supports object oriented programing and it is platform independent so ease to use and cross platform capabilities of java are impressive as in other languages like c it is platform dependent Java also provides different API and third party vendors support its library support is also good as comparison to lisp The resource management is also a plus point as we have to manage memory in c and other languages whereas here we have garbage collector so this is a plus point too Java is robust multithreaded and secure there are large communities of java developers and a lot of multifunctional ides

Java don’t have unassigned variables or long or short variables but we get these features in C/C++ so we can say that every language has its own perks and drawbacks