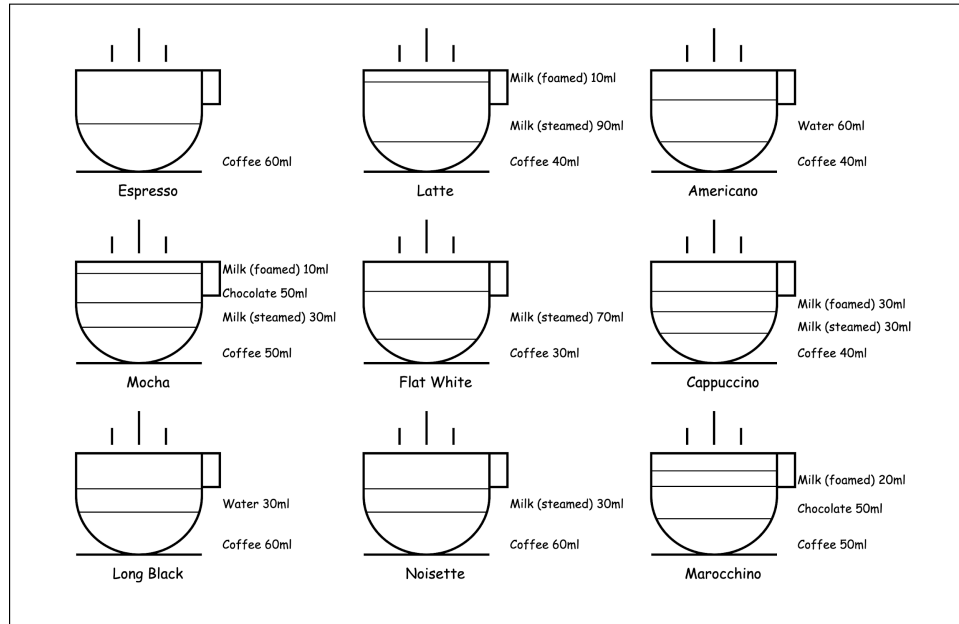


# Programming Recruitment Test

*Submission Deadline: 1 week*

Imagine that you are writing code for your cafe's coffee machine. The machine is so smart it is able to make whatever recipe you provide it, and here is the recipe/menu you have written:



There are 4 types of ingredients: Coffee, Milk, Water, and Chocolate. The machine is built with slots to contain each ingredient, with different volume size:

- Slot for coffee has total volume of 1000 ml
- Slot for milk has total volume of 1000 ml
- Slot for water has total volume of 500 ml
- Slot for chocolate has total volume of 300 ml

You need to write a web application for the user to operate the coffee machine. The application must provide interfaces to the following operations:

Interface	Input	Description
Display Menu		Display list of available drinks. A drink is said to be available if all of its ingredients supply allows to make at least 1 cup of the drink. For example, if the machine only has 10 ml of chocolate left in the slot, it should not include Mocha or Marocchino in the menu list, or indicate them as unavailable
Fill Ingredient Slot	ingredient , volume	Fill an ingredient into corresponding slot. The interface should reject (and show alert) if the volume of the filling ingredient would cause slot overflow

Order Drink	drink's name, number of cups	Start making the ordered drinks by taking the required ingredients out of the slots. Were the ingredients not sufficient, it should reject and show alert. The machine needs to correctly update the remaining ingredients
Display Current Supply	ingredient	Show the remaining supply of an ingredient (in its respective slot)
Display Stats		Show a report of the total number of cups ordered for each drink, and total ingredients consumed so far

#### Technical Specifications:

- Choose any language or framework which works best for you
- There are no restrictions on how to structure the pages; single page website is acceptable
- It is not necessary to store data permanently in database (local memory is enough), but feel free to do so if you like
- Write code in clean, structured, and efficient manner
- Use Git for version control
- Provide documentation (Word docs, diagrams, or simple README) which explained how you approach the problem and design the solution

#### Deliverables:

- Upload the source code (to Github or similar repository) and submit the URL
- Deploy the code in a remote server (e.g. Heroku) and submit the URL