

Pensamiento computacional para ingeniería  
Tecnológico de Monterrey Campus Chihuahua  
Eva Lucero Pérez Salcedo A01568830  
Víctor Manuel Puga R. A01568636  
21 de octubre del 2020



## Inventory python

### Final report

#### Tasks

- Register sales “Víctor”
- Register product arrivals “Eva”
- Query inventory data “Eva”
- Most sold items “Eva”
- Employees with most sales “Víctor”
- Generate employee's sales report “Víctor”

#### Proposed Options

- Show only seasonal products “Eva”
- Customer satisfaction form “Víctor”

### Screens

#### 0. Main Menu

Select an action

1. Register sale
2. Register product arrival
3. Query inventory data
4. Most sold items
5. Employees with most sales
6. Generate sales report
7. Show only seasonal products
8. Customer satisfaction form

Action: \_\_\_\_\_

→ Action: 2

## 1. Register Sale

Action: 1

--- Register sale ---

Who is selling the product?:

- Juan (1)
- Pedro (2)

Name or Id: \_\_\_\_\_

→ Name or Id: Pedro

→ Name or Id: 2

Which product is it?:

- Coffee (1) (3 in stock)
- Tea (2) (10 in stock)

Name or Id: \_\_\_\_\_

→ Name or Id: Coffee

→ Name or Id: 1

How many items? \_\_\_\_\_

→ How many items? 4

The order is valid. Calculating total price...

Total price: \$\_\_\_\_\_ (+ \_\_\_\_\_ tax)

This order's id is \_\_\_\_\_

Press enter to return to main screen

Press r to register another sale

## 2. Register Product Arrival

```
Action:2
--- Register product arrival ---

Which product is it?:
- Coffee (1)
- Tea (2)

Name or Id: _____      → Name or Id: Coffee
                             → Name or Id: 1

Recent arrival quantity: _____      → Recent arrival quantity:
123456

OKAY. You registered _____ items of _____ product
Now there are _____ in stock

Press enter to return to main screen
Press r to register another product
```

## 3. Query Inventory Data

```
Action: 3
--- Query inventory data ---

Which product is it?:
- Coffee (1)
- Tea (2)

Name or Id: _____      → Name or Id: Coffee
                             → Name or Id: 1

Information for _____

Description: _____
Id: _____
Price per unit: _____
Quantity in stock: _____
Season: _____
Categories: _____, _____

Press enter to return to main screen
Press r to search another product
```

#### 4. Most Sold Item

```
Action: 4
--- Most sold items ---

Top 3 most sold product until now:
1) _____ with _____ units sold
2) _____ with _____ units sold
3) _____ with _____ units sold

Press enter to return to main screen
```

#### 5. Show Employee with Most Items Sold

```
Action: 5
--- Show employees with most items sold ---

Top 3 employees:
1) _____ with _____ items sold
2) _____ with _____ items sold
3) _____ with _____ items sold

Press enter to return to main screen
```

#### 6. Generate Sales Report

```
Action: 6
--- Generate employee's sales report ---

Select an employee:
- Juan (1)
- Pedro (2)

Name or Id: _____      --> Name or Id: Pedro
                             --> Name or Id: 2

Creating file...
File saved as "_____"

Press enter to return to main screen
Press r to generate another report
```

**Sample output file**

EMPLOYEE NAME		
-----	-----	-----
PRODUCT ID	NAME	QUANTITY
PRODUCT ID	NAME	QUANTITY
PRODUCT ID	NAME	QUANTITY

**7. Show Only Seasonal Products**

```

Action: 7
--- Show only seasonal products ---

ALL 1
SPRING 2
SUMMER 4
FALL 3
WINTER 4

Select a season: ____

Products available only this season:
- ____
- ____
- ____

Press enter to return to main screen
Press r to search in another season

```

**8. Customer Satisfaction Form**

```

Action: 8
--- Customer satisfaction form ---

Which is your sale id (it is found on your receipt)? ____

How was our service? (1, 2, 3, 4, 5) ____

Cool. Thanks for giving us your feedback.
We hope to see you again.

Press enter to return to main screen

```

## Data Models

### Products

id	name	price	quantity	season	type	syb_type	description
int	string	float	int	string	string	string	string

```
product = {  
  "id": 0,  
  "name": "Frappe",  
  "price": 20.00,  
  "quantity": 234567,  
  "season": "ALL",  
  "type": "DRINK",  
  "sub_type": "COLD_COFFEE",  
  "description": "..."  
}
```

### Employees

id	name	last_name	position
int	string	string	string

```
employee = {  
  "id": 0,  
  "name": "John",  
  "last_name": "Appleseed",  
  "position": "MANAGER",  
}
```

## Sale

id	date	total_price	num_products	product_id	employee_id
int	string	float	int	int	int

```
sale = {  
    "id": 0,  
    "date": "18/01/2020",  
    "total_price": 300.00,  
    "num_products": 15,  
    "product_id": 0,  
    "employee_id": 0,  
}
```

## Feedback (Beta)

id	date	sale_id	rating
int	float	int	int

```
feedback = {  
    "id": 0,  
    "date": "18/01/2020",  
    "sale_id": 0,  
    "rating": 5,  
}
```

## Enumerations

Seasons		Rating		Positions		Types		Sub Types
"ALL "		1		"REGISTER"		"DRINK"		"HOT_COFFEE"
"SPRING"		2		"MANAGER"		"FOOD"		"HOT_DRINK"
"SUMMER"		3		"WAITER"				"HOT_DRINK"
"FALL "		4		"BARISTA"				"COLD_COFFEE"
"WINTER"		5						"COLD_DRINK"
								"COLD_DRINK"
								"BREAKFAST"
								"LUNCH"
								"SNACK"

### Retos relevantes

Lo más retador en nuestro proyecto fue la planeación de las páginas, así como la forma en que los archivos tendrían su lugar determinado. Cuando creamos las variables, en ocasiones coincidían con un nombre similar a otras y eso nos dificultó un poco el entendimiento de cómo lo estábamos haciendo, e incluso nos causaban errores sin saberlo. En algunas funciones complejas, cuando queríamos obtener los primeros lugares de las categorías de productos y empleados, hubo dificultades en como obtenerlos ya que se pensaba originalmente en una función, pero nos causaba demasiados errores por no conocer como realmente operaba.

### Solución de los retos

Al inicio fue difícil estructurar por dónde empezar, pero una vez hecha la planeación y como se iban a dividir los códigos, nos guiamos por ella. Para resolver las variables similares tuvimos que leer cuidadosamente como se estaban usando y en donde, revisando cada renglón que podría ser susceptible a fallos. En la función de los primeros



tres lugares, tuvimos que replantear como obtener esos resultados, pero con otra manera que éramos más familiar.

### **Reflexión sobre el aprendizaje durante el proyecto integrador**

La manera en que quisimos trabajar fue organizarnos como si se tratara de un proyecto de software en la vida real, dedicamos tiempos específicos varios días de la semana para ayudarnos en cada paso y supervisar el progreso. Aprendimos sobre la conexión entre archivos y funciones que facilitan la eficacia del programa al leerlo o editarlo. Además de que, al leer el documento de la planeación aprendimos a descifrar como transportarlo a una codificación.

La utilidad de estos programas para el uso diario en empresas será fundamental en un futuro no tan lejano, ya que el mundo está en constante cambio, se ha dependido de las tecnologías más que antes. Nuestro programa utilizado, Python, es solo una herramienta para la forma en que se pueden evolucionar estos servicios ya que incluye las funciones necesarias para realizar estos procesos y generar bases de datos para la automatización de los negocios.