

Evaluación Parcial N°3

Versión 2

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
FPY1101	Fundamentos de Programación	3 h	30%

1. Situación evaluativa

<input checked="" type="checkbox"/>	Ejecución práctica
-------------------------------------	--------------------

2. Agente evaluativo

<input checked="" type="checkbox"/>	Heteroevaluación	<input type="checkbox"/>	Coevaluación	<input type="checkbox"/>	Autoevaluación
-------------------------------------	------------------	--------------------------	--------------	--------------------------	----------------

3. Tabla de Especificaciones

Resultado de Aprendizaje	Indicador de Logro (IL)	Indicador de Evaluación (IE)	Ponderación Indicador Logro	Ponderación Indicador de Evaluación ¹
RA3 Construye estructuras de datos temporales, utilizando arreglos para almacenar datos, con el fin de darle persistencia a éstos y trabajar con problemas que necesiten guardar información temporalmente.	IL 3.1 Identifica colecciones y arreglos que permitan el almacenamiento de datos según los requerimientos del problema planteado.	IE 3.1.1 Identifica colecciones y arreglos que permitan el almacenamiento de datos según los requerimientos del problema planteado.	20%	20%
	IL 3.2 Utiliza arreglos y matrices para la inserción, eliminación, modificación y búsqueda de datos temporales para cumplir con los requerimientos del problema planteado.	IE 3.2.1 Utiliza arreglos y matrices para la inserción, eliminación, modificación y búsqueda de datos temporales para cumplir con los requerimientos del problema planteado.	20%	20%
	IL 3.3 Utiliza archivos para lograr la persistencia de los datos de la aplicación según el caso planteado.	IE 3.3.1 Utiliza archivos para lograr la persistencia de los datos de la aplicación según el caso planteado.	10%	10%
RA4 Modulariza programas, mediante funciones con y sin parámetros, para simplificar y darle eficiencia a las soluciones construidas.	IL 4.1 Utiliza librerías de sistema para la optimización del código según el caso planteado.	IE 4.1.1 Utiliza librerías de sistema para la optimización del código según el caso planteado.	15%	15%
	IL 4.2 Programa funciones que permitan la reutilización de código según el problema planteado.	IE 4.2.1 Programa funciones que permitan la reutilización de código según el problema planteado.	15%	15%
	IL 4.3 Integra funciones invocadas desde el programa principal para dar solución al problema planteado.	IE 4.3.1 Integra funciones invocadas desde el programa principal para dar solución al problema planteado.	10%	10%

¹ La ponderación del Indicador de Logro, debe ser distribuida como puntaje entre las preguntas que lo conformen.

	IL 4.4 Utiliza un sistema de control para administrar las versiones código (GitHub) en el desarrollo de una aplicación.	IE 4.4.1 Utiliza un sistema de control para administrar las versiones código (GitHub) en el desarrollo de una aplicación.	10%	10%
Total			100%	100%

4. Instrucciones generales para el/la estudiante

Esta es una evaluación que corresponde a una prueba de ejecución práctica y tiene un **30%** de ponderación sobre la nota final de la asignatura.

El **tiempo** para desarrollar esta evaluación es de **180 minutos** y se realiza de manera **individual** en **laboratorio**

La evaluación consiste en:

- Desarrollo de una aplicación en Python y en entorno de desarrollo Visual Studio Code
- Uso de colecciones
- Uso de archivos de texto
- Uso de librerías estándar de Python
- Uso de GitHub como repositorio del proyecto

5. Evaluación

Estimado/a estudiante,

Desarrolle una aplicación en Python utilizando Visual Studio que permita resolver el siguiente caso:

La empresa de catering "Gourmet Services" necesita desarrollar un sistema que permita gestionar los pedidos de sus clientes para eventos corporativos y privados. Para el funcionamiento del sistema se requiere las siguientes funcionalidades:

Registrar Pedido: Para registrar un pedido se requiere lo siguiente: Nombre y apellido del cliente, número de contacto, detalles del evento (tipo de evento, fecha y dirección), y menú seleccionado. Por ejemplo, la empresa ofrece menús de Comida Italiana, Comida Japonesa, y BBQ. Debe permitir seleccionar entre una de las 3 opciones e ingresar la cantidad de comensales. Por lo tanto, un detalle de pedido podría verse registrado de la siguiente manera:

Cliente	Contacto	Evento	Menú	Comensales
Juan Pérez	987654321	Reunión corporativa	Comida Italiana	20
María López	123456789	Cumpleaños	BBQ	50

Debe validar que todos los datos sean ingresados.

Listar Pedidos: Debe mostrar en la pantalla la lista de todos los pedidos realizados, similar al ejemplo anterior de registro de pedidos.

Imprimir Detalle de Pedidos por Menú: Para imprimir el detalle de pedidos, el usuario debe seleccionar alguno de los menús ofrecidos. Estos menús deben estar previamente definidos en algún tipo de colección de Python en el código, y por lo menos

deben ser tres, como los mencionados anteriormente. Al seleccionar uno de los menús se generarán 2 archivos de texto (.txt y un .json) con el detalle de los pedidos que se han realizado con ese menú específico. Este debe tener la misma forma del registro completo de las opciones anteriores pero en archivo de texto.

Salir del Programa: El programa debe funcionar hasta que el usuario decida salir.

GitHub: El código desarrollado por el estudiante debe ser subido en su plenitud a GitHub.