**Introducción.**

**P**unta sal es un balneario pequeño con buena infraestructura hotelera. La superficie de su playa es muy irregular, llegando la marea alta hasta el borde de las casas ubicadas al sur del balneario. Durante la marea baja quedan expuestas playas de arena fina.

Pues en el siguiente trabajo observaremos detalle a detalle las características del “Hotel Royal Decamerón Punta Sal”. Desde cómo hacer la reservación en el hotel hasta como hacer la reservación en los restaurantes.

Claro está que lo mencionado no es más que nuestro tema ejemplo, puesto que nuestro interés en este trabajo es utilizar programación orienta a objetos “POO” que utilizaremos para hacer un “[**Programa de Organización Hotel Royal Decamerón Punta Sal**](https://github.com/jpachecot/Programa_Organizacion_Hotel_Royal_Decameron_Punta_Sal)**”.**

La POO nos permite agrupar secciones de código con funcionalidades comunes.   
  
Una de las principales desventajas de la programación procedural basada en funciones es su construcción, cuando una aplicación bajo este tipo de programación crece, la modificación del código se hace muy trabajosa y difícil debido a que el cambio de una sola línea en una función, puede acarrear la modificación de muchas otras líneas de código pertenecientes a otras funciones que estén relacionadas.   
  
Con la programación orientada a objetos se pretende agrupar el código encapsulándolo y haciéndolo independiente, de manera que una modificación debida al crecimiento de la aplicación solo afecte a unas pocas líneas.

**Objetivos.**

* Los/as alumnos/as reconocerán los elementos fundamentales de un programa orientado a objetos (clase, objeto, estado, atributos, métodos y mensajes).
* Los/as alumnos/as deben ser capaces de definir clases de forma aislada y en el contexto de un programa orientado a objetos.
* Los/as alumnos/as reconocerán las diferentes relaciones que se establecen entre las distintas clases que forman un programa orientado a objetos (composición, uso y asociación).
* Los/as alumnos/as deben poder distinguir, utilizar, y definir relaciones de herencia entre clases en un programa orientado a objetos.
* catalogar y diferenciar el código, en base a estructuras jerárquicas dependientes, al estilo de un árbol genealógico.

Justificación.

**S**e propone crear un modelo que exponga la creación [**Programa Organización Hotel Royal Decamerón Punta Sal**](https://github.com/jpachecot/Programa_Organizacion_Hotel_Royal_Decameron_Punta_Sal). .

Hay que recordar que uno de los objetivos principales de la programación orientada a objetos es el de la reutilización del código, que en este caso son los objetos. Realizar una reingeniería de procesos no es una tarea simple y en ocasiones es preferible rehacer el módulo al momento en que el diseño no es claro.

Uno de los retos en este documento es el de proyectar una alternativa de programación mediante *un paradigma proactivo orientado a objetos*. Esta opción permitirá experimentar de manera formal con diferentes modelos. Además de desarrollar una herramienta formal de programación.

Marco teórico.

* **PyCharm**

[**PyCharm**](http://ift.tt/op9oPR)es un IDE o entorno de desarrollo integrado multiplataforma utilizado para[desarrollar](http://ift.tt/SalScR) en el lenguaje de programación Python. Proporciona análisis de código, depuración gráfica, integración con VCS / DVCS y soporte para el desarrollo web con Django, entre otras bondades. PyCharm es desarrollado por la empresa JetBrains y debido a la naturaleza de sus licencias tiene dos versiones, la Community que es gratuita y orientada a la educación y al desarrollo puro en Python y la Professional, que incluye más características como el soporte a desarrollo web con varios precios que van desde los 29$ al año.

* **Python**

Python es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad.   
  
En los últimos años el lenguaje se ha hecho muy popular, gracias a varias razones como:

* La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje, que ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.
* La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de 3 a 5 líneas de código menos que su equivalente en Java o C.
* La cantidad de plataformas en las que podemos desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otros.
* Además, Python es gratuito, incluso para propósitos empresariales.