Boletín 25

1. Para qué se utilizan os xenericos

A programación xenérica disminúe a cantidade de clases e métodos que se deben construir para cada aplicación que se desarrolle. Non recibe datos primitivos, polo que, é necesario traballar con tipos genéricos ou parametrizados

1. Podemos facer o mesmo ca clase object ¿de ser así, que ventaxas teñen os xenericos

Sirven para reutilizar código, reducir erros en tempo de execución xa que se detectan mentres se va compilando e disminúe a necesidade de facer conversións de tipos primitivos.

Object al ser tan general, debe emplear un casteo para saber a que tipo de dato trata. Otro punto é reducir os erros en tempo de execución, xa que o Object non dá error ata executar o programa

1. Que son os comodines?

Os comodines representan un tipo desconocido. Podese usar nunha variedade de situacións, como o tipo do parámetro, campo o variable local; ás veces como un tipo de devolución

1. Só podemos implementar métodos xenericos?

Non, pódese generar clases, interfaces e métodos que operan con tipos de datos parametrizados

1. Como indicamos que un método é xenerico?

Antes del retorno del dato se pone < T >

< T > → é un tipo de dato parametrizado o genérico (Diamante)

1. Cando debemos utilizar unha clase que implemente as seguintes interfaces?

* Set → Cando se quere un conxunto de elementos que non admita elementos duplicados
* Queue → Cando queremos ter unha cola na que a primeira entrada é a primeira en saír (FIFO)
* Map → Cando queremos representar unha estrutura de datos para almacenar pares clave-valor (non se repiten os valores) ; de tal maneira que para unha clave só temos un valor.
* Pila → Cando queremos ter unha cola na que o último elemento que entra vai a ser o primeiro en saír (LIFO)