

1. Con la ayuda del **IDE NetBeans** crear un proyecto de tipo **Aplicación Java**, llamado **AnalisisMayorModularArgumento**, que contenga su punto de entrada en la clase **Main** que se encuentra en el paquete **org.japo.java.main**, que llame al método **lanzarAplicacion** de la clase **App** que se encuentra en el paquete **org.japo.java.app**.

El programa debe de:

1. **Definir** las **constantes numéricas de clase N1 y N2** inicializadas con **sendos valores arbitrarios**.
 2. **Definir** un método llamado **devolverMayor**, que debe **recibir** dos argumentos llamados **n1** y **n2**, **analizar** la relación de magnitud entre ellos y **devolver** la magnitud del mayor de ellos, con la siguiente **signatura**:

```
private int devolverMayor(int n1, int n2);
```
 3. **Llamar** al método **devolverMayor**.
 4. **Mostrar** por pantalla un **mensaje informativo** del proceso en el siguiente formato:

```
El mayor de <n1> y <n2> es <nX>
```
2. Con la ayuda del **IDE NetBeans** crear un proyecto de tipo **Aplicación Java**, llamado **EntradaEnteroModularArgumento**, que contenga su punto de entrada en la clase **Main** que se encuentra en el paquete **org.japo.java.main**, que llame al método **lanzarAplicacion** de la clase **App** que se encuentra en el paquete **org.japo.java.app**.

El programa debe de:

1. **Definir** la **constante de clase MSG_USR** inicializada con el mensaje que debe indicarse al usuario para invitarle a que introduzca un número por teclado.
2. **Definir** la **constante de clase MSG_ERR** inicializada con el mensaje que debe indicarse al usuario para informarle de que ha habido un error en el proceso de introducción de un número por teclado.
3. **Definir** un método llamado **obtenerEntero**, que debe gestionar la **petición al usuario** de un **número entero** y su posterior **devolución**, en el bien entendido de que en caso de error se debe de repetir la petición, con la siguiente **signatura**:

```
private int obtenerEntero(String msgUsr, String msgErr);
```
4. **Llamar** al método **obtenerEntero** para **obtener el número introducido por el usuario**.
5. **Informar** finalmente del número introducido por el usuario en el siguiente formato:

```
Número introducido ...: <número>
```

3. Con la ayuda del **IDE NetBeans** crear un proyecto de tipo **Aplicación Java**, llamado **AnalisisRangoModularArgumento**, que contenga su punto de entrada en la clase **Main** que se encuentra en el paquete **org.japo.java.main**, que llame al método **lanzarAplicacion** de la clase **App** que se encuentra en el paquete **org.japo.java.app**.

El programa debe de:

1. **Definir** la **constante de clase NUM** inicializada con el valor correspondiente a un número arbitrario.
2. **Definir** las **constantes de clase MIN y MAX** inicializadas con los valores correspondientes a un rango arbitrario.
3. **Definir** un método llamado **analizarEntero**, que recibe un número y los límites de un rango, y que debe de devolver un valor lógico que codifique si el número pasado está **dentro** del rango [**true**] o **fuera** del rango [**false**], con la siguiente **signatura**:

```
private boolean analizarEntero(int num, int min, int max);
```

4. **Llamar** al método **analizarEntero** con las constantes definidas previamente para **obtener el estado de pertenencia del número al rango**.
5. **Informar** del estado de pertenencia del número al rango, en el formato:

```
<num> está (DENTRO|FUERA) del rango [<min>..<max>]
```

4. Con la ayuda del **IDE NetBeans** crear un proyecto de tipo **Aplicación Java**, llamado **TestDNIModularArgumento**, que contenga su punto de entrada en la clase **Main** que se encuentra en el paquete **org.japo.java.main**, que llame al método **lanzarAplicacion** de la clase **App** que se encuentra en el paquete **org.japo.java.app**.

El programa debe de:

1. **Definir** la **constante de clase LETRAS** inicializada con la **secuencia necesaria**.
2. **Definir** la **constante de clase DNI_NUMERO** inicializada con un **número entero** correspondiente a un DNI arbitrario.
3. **Definir** la **constante de clase DNI_CONTROL** inicializada con un carácter alfabético correspondiente a un DNI arbitrario.
4. **Definir** un método llamado **comprobarLetraDNI**, que debe analizar y devolver el estado de correspondencia del numero y la letra suministrados a un **DNI correcto** [**true**] o **DNI incorrecto** [**false**], con la siguiente **signatura**:

```
private boolean comprobarLetraDNI(int num, char ctr);
```

5. **Definir** un método llamado **calcularLetraDNI**, que debe calcular y devolver la letra de control asociada al número de DNI suministrado, con la siguiente **signatura**:

```
private char calcularLetraDNI(int num);
```

6. **Llamar** al método **comprobarLetraDNI** con las constantes **DNI_NUMERO** y **DNI_CONTROL** para averiguar si se corresponden con un DNI correcto o no.
7. **Mostrar** el resultado del análisis por pantalla en el **formato**:
E1 DNI NNNNNNNNL (SI|NO) es correcto.