

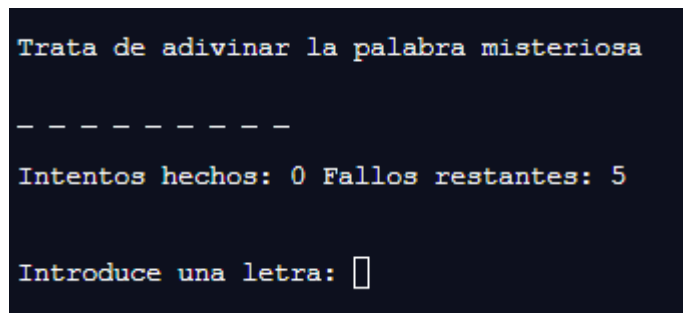
Realiza un programa en Java que rete al usuario a adivinar una palabra entre las que forman parte de su diccionario. La palabra a adivinar debe ser elegida aleatoriamente en cada ejecución.

Al usuario se le deben mostrar enmascarados los caracteres que forman parte de la palabra requerida al principio de la ejecución y se deben ir desvelando según se vayan acertando en cualquier momento del desarrollo de la adivinanza. También habrá que informar del número de intentos que se llevan hechos y de las oportunidades de fallar que le quedan.

El programa terminará por dos motivos: si se consumen todos los fallos permitidos sin desvelar completamente la palabra se habrá fallado en la adivinanza, o bien, se acierta la palabra antes de gastar todos los fallos tolerados.

Tras acabar, exitosamente o no, hay que mostrar un mensaje por pantalla con esta circunstancia y la palabra misteriosa que se proponía para adivinar.

El aspecto de algunas capturas de pantalla es el siguiente:



```
Trata de adivinar la palabra misteriosa
- - - - -
Intentos hechos: 0 Fallos restantes: 5
Introduce una letra: □
```

Pantalla inicial

```
Introduce una letra: f
- - - - -
Intentos hechos: 1 Fallos restantes: 4

Introduce una letra: o
- - - - -
Intentos hechos: 2 Fallos restantes: 3

Introduce una letra: a
_ a _ a _ _ _ a
Intentos hechos: 3 Fallos restantes: 3

Introduce una letra: 
```

Varios intentos realizados

```
Introduce una letra: s
_ a _ a s _ _ _ a
Intentos hechos: 4 Fallos restantes: 3

Introduce una letra: h
_ a _ a s _ _ _ a
Intentos hechos: 5 Fallos restantes: 2

Introduce una letra: e
_ a _ a s _ e _ a
Intentos hechos: 6 Fallos restantes: 2
```

Tras más intentos hechos

```
Introduce una letra: m
_ a _ a s _ e _ a
Intentos hechos: 7 Fallos restantes: 1

Introduce una letra: j

La palabra misteriosa era: canastera
Lo siento, has fallado tras 8 intentos✖
```

Fin de ejecución sin acierto

```
Introduce una letra: c
v e n c e _ o
Intentos hechos: 9 Fallos restantes: 1

Introduce una letra: j

La palabra misteriosa era: vencejo
ENHORABUENA, has acertado con 10 intentos✔
```

Fin de otra ejecución con acierto del usuario

#### Aclaraciones:

- El número de fallos permitidos será de cinco.
- El diccionario de palabras entre las que extraer la que hay que adivinar será el String "*palabras*", cuyo contenido es:

```
String palabras = " golondrina avetoro codorniz garcilla alcotan abubilla  
perdiz abejaruco vencejo gavilan papamoscas petirrojo colirrojo pechiazul  
lavandera mosquitero milano aguilas roquero estornino treparriscos halcon buitre  
canastera picapinos gorrion verdon verdecillo buitron bigotudo "; // 30  
valores distintos
```

- La instrucción para generar un número aleatorio entre los valores que se requieren es: `numeroAleatorio = (int) (Math.random() * numPalabras);` donde *numPalabras* toma el valor 29 porque hay 30 valores y así esa instrucción va a dar un número aleatorio de 0 a 29.
- Hay que tener en cuenta que "A" y "a" son caracteres distintos, por eso para tratar de simplificar la resolución de la práctica el String proporcionado no contiene mayúsculas ni tildes.
- La instrucción para leer un carácter con Scanner es: `sc.next().charAt(0);`