



Área personal 2018-2019 Desarrollo de aplicaciones web 1819_DAW_PROG_EE Acceso a BBDD: JDBC (10%)

PROG10 Tarea de evaluación 01. Realiza un programa en Java (100%)

Administración



Administración del curso

PROG10 Tarea de evaluación 01. Realiza un programa en Java (100%)

ENUNCIADO.



En esta clase crearemos los siguientes métodos:

```
public String getBBDD()
public String[] arrayEquipos()
public ArrayList<Jugador> arrayListJugadores()
public String[] arrayProcedencia()
String selectJugadores(String procedencia, String posicion)
public String selectEstadisticas(String equipoElegido)
public boolean insertarPartido(String local, String visitante, int ptosLocal, int ptosVisitante, String temporada)
)
public boolean cambiarJugador(String nombre, String altura, int peso, String posicion, String equipo)
```

Dentro del fichero **ClaseBBDD.java** puedes obtener más información sobre cada uno de estos métodos. Todos ellos están comentados para que el fichero compile correctamente.

Las tablas con las que vamos a trabajar son de la tarea de aprendizaje 3 de BD05 y están en la máquina virtual con la que estamos trabajando. Para acceder a ellas tendremos que usar el usuario nba y la contraseña nba. Tanto el usuario como la contraseña y la ip que vamos a usar se guardarán en el fichero **config.dat** y se leerán al principio del main.

En nuestras consultas trabajaremos con 4 tablas.

La tabla equipos tendrá 4 campos (nombre, ciudad, conferencia y división) y almacenará información de los equipos que van a participar en la competición.

	NOMBRE	CIUDAD	CONFERENCIA	DIVISION
1	Celtics	Boston	East	Atlantic
2	Raptors	Toronto	East	Atlantic
3	76ers	Philadelphia	East	Atlantic
4	Nets	New Jersey	East	Atlantic

La tabla jugadores tendrá 7 campos (codigo, nombre, procedencia, altura, peso, posición y nombre_equipo) y almacenará la información de todos los jugadores de la liga. Estará unida a la tabla equipos a través del campo nombre_equipo.

	CODIGO	NOMBRE	PROCEDENCIA	ALTURA	PESO	POSICION	NOMBRE_EQUIPO
1	1	Corey Brever	Florida	6-9	185	F-G	Timberwolves
2	2	Greg Buckner	Clemson	6-4	210	G-F	Timberwolves
3	3	Michael Doleac	Utah	6-11	262	C	Timberwolves
4	4	Randy Foye	Villanova	6-4	213	G	Timberwolves

La tabla estadísticas tendrá 6 campos (temporada, jugador, puntos_por_partido, asistencias_por_partido, tapones_por_partido y rebotes_por_partido) y almacenará las de los jugadores durante las diferentes temporadas. Estará unida a la tabla jugadores a través del campo jugador que coincidirá con su código.

	TEMPORADA	JUGADOR	PUNTOS_POR_PARTIDO	ASISTENCIAS_POR_PARTIDO	TAPONES_POR_PARTIDO	REBOTES_POR_PARTIDO
1	03/04	283	5,9	2,2	1,54	1,6
2	04/05	283	4,6	1,8	0,98	1,6
3	05/06	283	12	4,7	2,4	2,9
4	06/07	283	4,9	1,3	0,89	0,8

La tabla partidos tendrá 6 campos (codigo, equipo_local, equipo_visitante, puntos_local, puntos_visitante y temporada) y almacenará la información de los partidos de cada temporada. Estará doblemente unida a la tabla equipos a través de los campos equipo_local y equipo_visitante.

	CODIGO	EQUIPO_LOCAL	EQUIPO_VISITANTE	PUNTOS_LOCAL	PUNTOS_VISITANTE	TEMPORADA
1	1	Raptors	Lakers	74	117	98/99
2	2	Raptors	Grizzlies	70	118	98/99
3	3	Raptors	Clippers	105	76	98/99
4	4	Raptors	Knicks	62	124	98/99

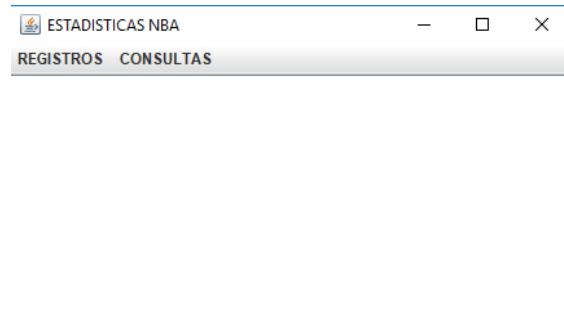
La interfaz gráfica necesita las clases **InterfazGrafica**, **DialogoInsertar**, **DialogoModificar**, **DialogoEstadisticas**, y **DialogoJugador**. En estos ficheros únicamente tendremos que añadir el código necesario para realizar las tareas indicadas. La mayoría de ellas nos permitirán integrar consultas a la BBDD en nuestra aplicación.

Se entregará un fichero .ZIP con todos los ficheros del proyecto.

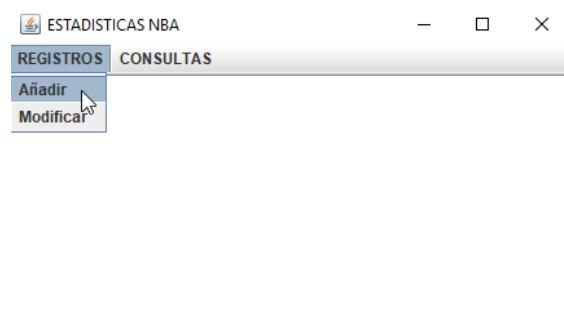


El programa comenzará leyendo la ip, el usuario y la contraseña para acceder a la BBDD del fichero config.dat. Con esa información ya podremos crear un objeto de la clase ClaseBBDD y otro de la clase InterfazGrafica.

La **ventana principal** de la aplicación tendrá el siguiente aspecto:



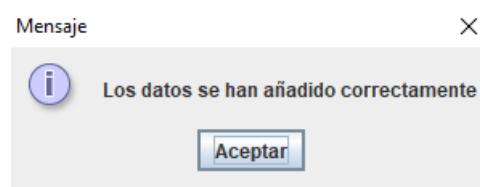
El menú "**REGISTROS**" tendrá 2 opciones:



Si elegimos "**Añadir**" se abrirá una nueva ventana en la que podremos introducir los datos de un nuevo partido:



Cuando pulsemos el botón "**AÑADIR**" se intentará introducir esos datos en la BBDD y se informará si la operación se ha realizado con éxito:



Si elegimos la opción "**Modificar**" se abrirá una nueva ventana en la que podremos elegir el jugador cuyos datos queremos cambiar:

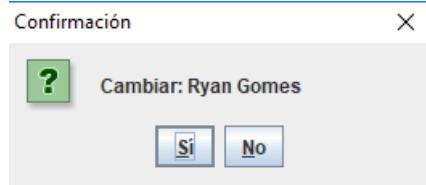


Cuando elegimos un jugador, se cargarán sus datos en las cajas de texto:

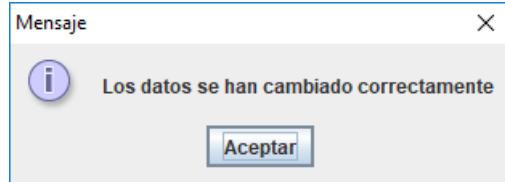


Jugador Ryan Gomes (Timberwolves)			
Altura:	6-7	Peso:	250
Posición:	F	Equipo:	Timberwolves
CAMBIAR			

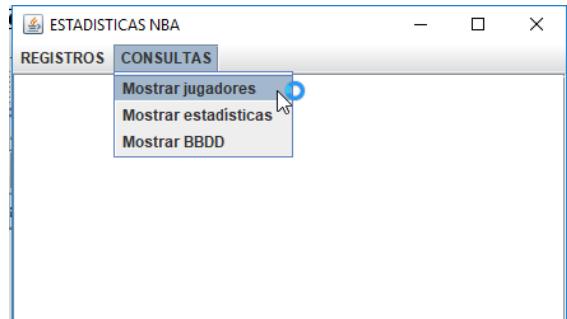
Al pulsar el botón "**CAMBIAR**" nos pedirá confirmación:



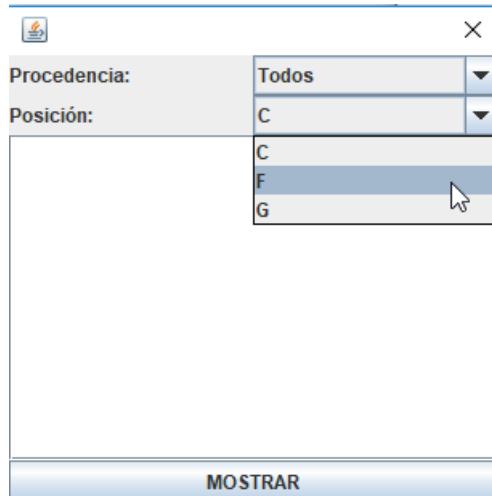
Si aceptamos, tratará de modificar los datos e informará si la operación se ha realizado con éxito:



El menú "**CONSULTAS**" tendrá 3 opciones.



Si elegimos "**Mostrar jugadores**" se abrirá una nueva ventana en la que podremos elegir la procedencia y la posición de un jugador:



En el área de texto se visualizarán los jugadores que cumplen con las condiciones elegidas:

Procedencia:	Todas	
Posición:	C	
MOSTRAR		
NOMBRE	PROCEDENCIA	POSICION
Michael Doleac	Utah	C
Al Jefferson	Prentiss Hs	C-F
Mark Madsen	Stanford	C-F
Craig Smith	Boston College	F-C
Chris Kaman	Central Michigan	C
Josh Powell	North Carolina State	C-F
Kwame Brown	Glynn Academy	C
Jason Collins	Stanford	C-F
Darko Milicic	Serbia Montenegro	C-F



Si elegimos "**Mostrar estadísticas**" se abrirá una nueva ventana en la que podremos elegir el equipo de cuyos jugadores queremos ver las estadísticas:



En el área de texto de la ventana principal se visualizarán las estadísticas de los jugadores de ese equipo:

ESTADISTICAS NBA		
REGISTROS	CONSULTAS	X
NOMBRE	REBOTES	TAPONES
Delonte West	3.13	0.5
Daniel Gibson	1.95	0.15
Zydrunas Ilgauskas	7.85	1.68
Devin Brown	3.12	0.18
Eric Snow	2.32	0.12
LeBron James	6.9	0.78
Joe Smith	6.68	0.91
Aleksandar Pavlovic	1.9	0.14
Anderson Varejao	6.18	0.55

Por último, si elegimos "**Mostrar BBDD**" se mostrará la información de la BBDD con la que estamos trabajando en un JOptionPane:



CRITERIOS DE PUNTUACIÓN. TOTAL 10 PUNTOS.



	Muy bueno (10)	Español - Internacional (es) Bueno (7-9)	Satisfactorio (5-7)	Jose Maria Tome Mejias Debe mejorar (< 4)
Funcionalidad (70%)	<p>El programa obtiene la ip, el usuario y la contraseña para acceder a la BBDD desde el fichero config.dat.</p> <p>Todas las opciones del menú funcionan correctamente.</p> <p>Se han añadido al menos 2 mejoras al programa.</p>	<p>El programa obtiene la ip, el usuario y la contraseña para acceder a la BBDD desde el fichero config.dat.</p> <p>De las 5 opciones del menú, 4 funcionan correctamente.</p> <p>Se han añadido al menos 2 mejoras al programa.</p>	<p>El programa obtiene la ip, el usuario y la contraseña para acceder a la BBDD desde el fichero config.dat.</p> <p>De las 5 opciones del menú, 2 funcionan correctamente.</p> <p>Se ha añadido al menos una mejora al programa.</p>	No cumple las especificaciones básicas

Comentarios (15%)	Están bien escritos, en los lugares apropiados y ayudan a entender el funcionamiento del programa	Se detecta código que se beneficiaría de un comentario o algunos comentarios son excesivos o inapropiados	Incluye la cabecera y las secciones principales están comentada	Faltan los comentarios mínimos
Legibilidad (15%)	El código está bien organizado, sigue la guía de estilo y es fácil de leer	El código está organizado pero se detectan incumplimientos menores (espacios en blanco, algún identificador) del código de estilo	El código está organizado pero se detectan incumplimientos importantes (indentación) del código de estilo	El código está mal organizado y es difícil de leer

Con esta rubrica se calculará la nota de vuestro programa y se comparará con la que habéis indicado en la autoevaluación.

- Si la diferencia es menor de un punto, se calculará la nota media entre las dos.
- Si la diferencia es mayor, se utilizará directamente la calificación de la profesora.

En los casos en los que falte la autoevaluación, se utilizará la calificación de la profesora con una penalización de un punto.

RECURSOS NECESARIOS PARA REALIZAR LA TAREA.

Para realizar esta tarea puedes utilizar jGRASP o el editor VPL de Moodle que además te dirá si la salida del programa coincide con la propuesta.

CONSEJOS Y RECOMENDACIONES.



Realiza el programa por partes. Escribe el código poco a poco compilándolo y testeándolo frecuentemente.

Puedes aprovechar el código que hemos visto en los diferentes ejemplos y modificarlo para adaptarlo a la aplicación.

1. Empieza por **leer el fichero config.dat** dentro del main para obtener todos los parámetros que necesitamos para trabajar con la BBDD. Con esa información ya podremos crear un objeto de la clase ClaseBBDD operativo y otro de la clase InterfazGrafica.
2. Crea el **método getBBDD** dentro de la clase ClaseBBDD y desarrolla la **opción "Mostrar BBDD"** en el fichero InterfazGrafica.java.
3. A partir de aquí, puedes **elegir el orden** de las opciones que quieras desarrollar.
4. Crea los **métodos arrayEquipos e insertarPartido** dentro de la clase ClaseBBDD y desarrolla la **opción "Añadir"** en el fichero DialogoInsertar.java.
5. Crea los **métodos arrayListJugadores y cambiarJugador** dentro de la clase ClaseBBDD y desarrolla la **opción "Modificar"** en el fichero DialogoModificar.java.
6. Crea los **métodos arrayProcedencia y selectJugadores** dentro de la clase ClaseBBDD y desarrolla la **opción "Mostrar jugadores"** en el fichero DialogoJugador.java.
7. Crea el **método selectEstadisticas** dentro de la clase ClaseBBDD y desarrolla la **opción "Mostrar estadísticas"** en el fichero DialogoEstadisticas.java.

Utiliza métodos, parámetros y returns para estructurar el programa y evitar redundancias.

Para resolver el programa puedes utilizar constantes.

Guía de estilo:

1. Da nombres significativos a métodos, variables y parámetros siguiendo además las reglas de Java.
2. Utiliza correctamente la indentación y los espacios en blanco.
3. Incluye cabeceras explicativas al principio del programa y al principio de cada uno de los métodos, así como junto a cualquier línea de código que lo pueda necesitar.
4. En el caso de los métodos, como comentario, además de lo que hace indica los parámetros que utiliza y el valor que devuelve si es que existe.

INDICACIONES DE ENTREGA.

El programa se escribirá y se testeará en el IDE personal.

En la cabecera, además de la información que describe el programa, se añadirá el enlace a la evaluación realizada, un enlace a Google Drive o a Youtube según el formato elegido.

Se comprimirán todos los ficheros utilizados en un único .ZIP.

Se entregará el fichero .ZIP.

Estado de la entrega

Estado de la entrega	No entregado
Estado de la calificación	No calificada
Fecha de entrega	martes, 21 de mayo de 2019, 23:55
Tiempo restante	15 días 3 horas
Última modificación	-
Comentarios de la entrega	Comentarios (0)

Agregar entrega