PRÁCTICA 3

Tecnología de la Programación

Ingeniería Informática 2007/2008

Resumen

En esta práctica continuaremos con el desarrollo del simulador de una pequeña entidad bancaria llamada MiniBank. Se trabajará en el desarrollo de un banco de pruebas que nos ayude a validar el correcto funcionamiento de nuestro programa y detectar los errores de funcionamiento, así como documentar correctamente las clases y métodos.

Objetivos

- Documentar de forma correcta el código fuente.
- Definir un conjunto de datos y baterías de pruebas.
- Comprender el concepto de cobertura de código.
- Utilizar **ant** para la compilación del programa y la ejecución de las pruebas (OPCIONAL)

Material

Para la realización de esta práctica partiremos del código desarrollado durante la práctica 2, se adjuntan nuevas versiones de algunas de las clases para permitir la ejecución del interprete de forma no interactiva. Las nuevas clases se encuentra subidas en la herramienta de teleformación utilizada por la asignatura.

Documentación adicional

Transparencias Javadoc

http://www.madsgroup.org/docencia/mod/resource/view.php?id=1036

Transparencias Ant

http://www.madsgroup.org/docencia/mod/resource/view.php?id=1041

Descripción del trabajo a realizar

Para la realización de esta práctica se partirá del resultado final de la práctica anterior P2, para ello se copiará (svn copy) el directorio P2 de la práctica anterior en un directorio P3.

La primera parte de la práctica consistirá en documentar siguiendo las normas el código fuente utilizando JavaDoc.

Se sustituirán los ficheros con las nuevas clases proporcionadas, para la interfaz de usuario, así como una nueva versión del código de la clase **FactoriaMinibank**. En esta clase se proporciona un nuevo método *public static IMiniBank crearMiniBank(String nombre)* que permite la creación de distintos MiniBank previamente inicializados.

Para la definición de los bancos de prueba iniciales, esta clase se debe modificar para poder utilizar bancos con datos iniciales ya precargados (Clientes, Cuentas, Operaciones).

La utilización de estos bancos será posible utilizando un parámetro con el nombre del Banco que se quiera cargar (BancoA,BancoB,BancoC). Es misión del alumno definir tres conjuntos de datos iniciales para la batería de pruebas de validación que se realizarán como parte de esta práctica. Sirva de ejemplo el siguiente código:

```
if (nombre.equals("BancoA")) {
        miniBank.crearCliente(“76000000R”,”Juan”,”Lopez Gomez”,”555123232”);
        miniBank.crearCuenta(“76000000R”,5000);
        // ESTO ES UN EJEMPLO DE DATOS BASE PARA PRUEBAS
        return miniBank;
}
```

La nueva versión del **Main** proporcionada permite una ejecución del entorno no interactiva, de forma que será posible la utilización de ficheros para realizar conjuntos de operaciones definidas en los mismos. Para ello habrá que pasarle el siguiente parámetro “-n” al llamar a Main.

El siguiente ejemplo ejecutaría de forma no interactiva el bancoA con los datos de .
$java Main -n bancoA

El alumno deberá diseñar y definir baterías de pruebas que realicen una cobertura del 100% de las líneas de código desarrolladas por el alumno. Una batería de pruebas queda definida en el entorno de esta práctica como el conjunto formado por los datos iniciales, una serie de ordenes almacenadas en un fichero de texto que se cargarán por la entrada estándar y el resultado esperado de la ejecución del código. Y el comando que ejecuta la prueba:

$java Main -n bancoA < Bateria1Ordenes.txt > Bateria1Salida.txt

Para obtener la cobertura del 100% el alumno solo podrá ejecutar baterías contra el interprete de comandos, se podrán detectar métodos que no son nunca utilizados, o

ramas de condiciones que no son nunca ejecutadas. O funcionamientos incorrectos o no contemplados de la aplicación.

Para que los resultados puedan ser comparables y evaluables es necesario que sea homogénea la presentación por pantalla de los resultados, utilizando una función toString común para todos:

---

*DATOS CLIENTE:
Nombre: XXXXXXXXXXXXXXXXXX
Apellidos: XXXXXXXXXXXXXXXXXX
Telefono: XXXXXXXXXXXXX
DNI: XXXXXXXXXXXXX

---

*DATOS CUENTA:
Numero: XXXXXXXXXXXXXXX
Titular: XXXXXXXXXXXXX
Saldo: XXXXXXX

---

*DATOS OPERACION:
Fecha: dd/mm/aaaa
Tipo: INGRESO|REINTEGRO
Importe: XXXXXXXXXXX

---

*DATOS OPERACION
Fecha: dd/mm/aaaa
Tipo: TRANSFERENCIA
Importe: XXXXXXXXXXX
Cuenta Origen: XXXXXX
Cuenta Destino: XXXXX

La notación obliga que después de cada dos puntos siempre hay un espacio, y no se acentúan las palabras.

En la memoria debéis describir los resultados obtenidos con el plan de pruebas, se recomienda ir documentando a medida que vais tomando decisiones, se considera un éxito que hayáis encontrado errores en el funcionamiento de vuestra práctica. Debéis documentar cada vez que encontréis un error, cada error lo debéis documentar de la siguiente forma y numerar de forma correlativa.

ERROR NUMERO XX:
Batería de pruebas: bateriaXXX
Fecha detección: dd/mm/aaaa
Descripción error: XXXXXXXXXXXXXXXXXXXXXXX
Descripción solución:
Validada batería de pruebas: (SI/NO)

Los errores detectados (BUGS) los iréis anotando en un fichero errores.txt en el directorio P3/tests/errores.txt, que subiréis al Subversión cada vez que realicéis un cambio o detectéis un error.

Cuando detectéis un error guardar los resultados de esa prueba, corregir el programa y repetir la prueba con los mismos datos y comprobar que la se ha solucionado el problema.

Entregables:

- El código documentado de todas las clases MiniBank.
- Las distintas baterías de pruebas (fichero de entrada, fichero de ejecución, fichero resultado ejecución y fichero describiendo los objetivos de la batería de pruebas.
- El fichero build.xml para la parte opcional de ANT.
- Memoria describiendo el trabajo realizado detallando como se han inicializado los diferentes Bancos de pruebas, así como el proceso de detección de errores y su corrección.
- Documentación javadoc generada en la carpeta doc/javadoc/

La entrega de las prácticas se realizará a través del repositorio Subversion de cada grupo de
prácticas, con la siguiente estructura:
/trunk/P3

src/ (código fuente)
test/bateria1 (fichero entrada, fichero resultante, descripción de pruebas realizadas y que partes del código cubre la batería de pruebas)
test/bateria2
.
test/bateriaN
doc/tp-gXXX-p3-memoria.pdf (donde XXX es el número de grupo)
doc/errores.txt (documentación detección de BUGs)
doc/javadoc/ (documentación javadoc generada)

Plazo de entrega

La duración estimada para la realización de la práctica es de dos semanas lectivas, por tanto la fecha de entrega es antes del 20/04/2008.