

Prueba Periodo de Recuperación SVN
------------------------------------

ESTEFANÍA GARCÍA SÁNCHEZ

<https://svn.code.sf.net/p/examensvnene21/code/>

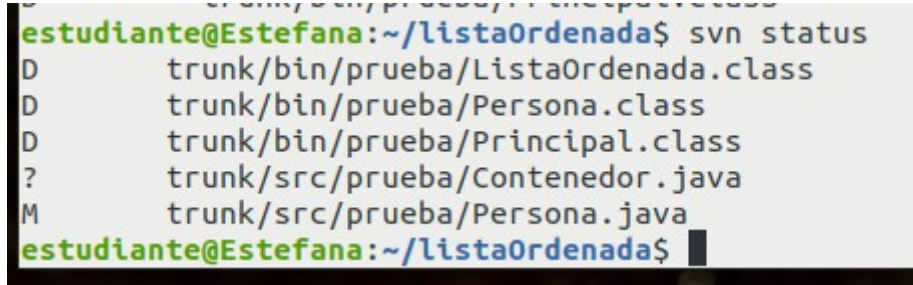
1. Crea un repositorio en sourceforge y dame acceso.
  1. `svn checkout --username=egarsan176`  
`https://svn.code.sf.net/p/examensvnene21/code/ examensvnene21-code`
2. En un directorio de la misma máquina, crea un directorio que representará un proyecto de desarrollo y añade los ficheros que te ha proporcionado el profesor en el fichero ListaOrdenadaGenerica.zip
  1. `mkdir -p examenSVN examenSVN/trunk examenSVN/branches examenSVN/tags`
  2. `mv Downloads/ListaOrdenadaGenerica/bin examenSVN/trunk/`
  3. `mv Downloads/ListaOrdenadaGenerica/src examenSVN/trunk/`
  4. `mv Downloads/ListaOrdenadaGenerica/.settings/ examenSVN/trunk/`
  5. `mv Downloads/ListaOrdenadaGenerica/.classpath examenSVN/trunk/`
  6. `mv Downloads/ListaOrdenadaGenerica/.project examenSVN/trunk/`
3. Importa ese proyecto que acabas de crear en el punto 2 al repositorio que creaste en el punto 1. Procura que el repositorio tenga la estructura recomendada en svn.
  1. `svn import /home/estudiante/examenSVN/`  
`https://svn.code.sf.net/p/examensvnene21/code/ -m "importado el proyecto de desarrollo"`
4. Obtén una copia de trabajo dentro del directorio listaOrdenada.
  1. `svn co https://svn.code.sf.net/p/examensvnene21/code/ listaOrdenada`
5. Modifica el método `compareTo` de la clase `Persona` (líneas 89 a 93 del fichero `Persona.java`) y cámbialo por este:

```
@Override
public int compareTo(Persona o) {
    return Integer.compare(edad, o.edad);
}
```

  1. `nano trunk/src/prueba/Persona.java`
6. Añade la clase `Contenedor.java` que te ha proporcionado el profesor.
  1. `Cd ..`
  2. `mv /home/estudiante/Downloads/Contenedor.java /home/estudiante/listaOrdenada/trunk/src/prueba/`
7. Elimina los ficheros compilados (todos los ficheros dentro del directorio `bin/prueba`, son los ficheros `.class`)
  1. `cd listaOrdenada`
  2. `svn delete trunk/bin/prueba/ListaOrdenada.class trunk/bin/prueba/Persona.class trunk/bin/prueba/Principal.class`

8. Mira el estado de la copia de trabajo y comprueba los códigos asociados a los archivos modificados.

1. Svn status



```
estudiante@Estefana:~/listaOrdenada$ svn status
D      trunk/bin/prueba/ListaOrdenada.class
D      trunk/bin/prueba/Persona.class
D      trunk/bin/prueba/Principal.class
?      trunk/src/prueba/Contenedor.java
M      trunk/src/prueba/Persona.java
estudiante@Estefana:~/listaOrdenada$
```

D → el ítem ha sido borrado

? → el ítem no está bajo el control de versión

M → el ítem ha sido modificado

9. Añade al control de versiones los archivos nuevos y elimina los archivos borrados. Crea una nueva revisión con los cambios realizados.

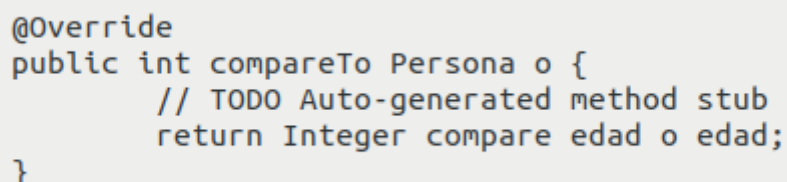
1. svn add trunk/src/prueba/Contenedor.java
2. svn ci -m "eliminados los archivos compilados y añadido Contenedor.java"

10. Mira en qué revisión se encuentra la copia de trabajo y mira todos los registros.

1. Svn status -u
2. svn diff -r 1:2 // svn log

11. Queremos que la clase Persona vuelva al comparador anterior.

1. svn merge -r 2:1 <https://svn.code.sf.net/p/examensvnene21/code/trunk/src/prueba/Persona.java>  
/home/estudiante/listaOrdenada/trunk/src/prueba
2. Me dice que no se puede hacer el merge, que antes haga un update. Ejecuto svn update y repito de nuevo el comando anterior
3. Para comprobar que ha vuelto a su versión anterior hago un cat trunk/src/prueba/Persona.java y efectivamente está sin modificar



```
@Override
public int compareTo(Persona o) {
    // TODO Auto-generated method stub
    return Integer.compare(edad, o.edad);
}
```

12. Manda una nueva revisión al repositorio.

1. svn ci -m "nueva versión con el archivo Persona.class revertido a su estado inicial"

13. Repite el punto 10. ¿Qué ha ocurrido?.

1. No ha ocurrido nada, el repositorio sigue en la versión dos porque lo que se ha traído al local la versión anterior de un fichero solamente.
14. Actualiza la copia de trabajo y repite el punto 10. ¿Qué ha ocurrido?
  1. Svn update
  2. svn status
  3. svn log
  4. Si miramos con cat el fichero vemos que ha vuelto a estar como tras su primera modificación
15. Recupera uno de los ficheros compilados que eliminaste en el punto 7, asegúrate que se recupere también en el repositorio.

1. svn copy  
`https://svn.code.sf.net/p/examensvnene21/code/trunk/bin/prueba/Principal.class@1`  
trunk/bin/prueba

```
estudiante@Estefana:~/listaOrdenada$ svn copy https://svn.code.sf.net/p/examensvnene21/code/trunk/bin/prueba/Principal.class@1 trunk/bin/prueba
A      trunk/bin/prueba/Principal.class
estudiante@Estefana:~/listaOrdenada$
```

16. Crea una rama privada de desarrollo sobre el directorio branches del proyecto. Llámala canelaEnRama.
  1. svn cp <https://svn.code.sf.net/p/examensvnene21/code/trunk/>  
<https://svn.code.sf.net/p/examensvnene21/code/branches/canelaEnRama> -m  
"creada la nueva rama de trabajo"
17. Haz una copia de trabajo de dicha rama en el directorio canela.
  1. Cd ..
  2. svn co <https://svn.code.sf.net/p/examensvnene21/code/branches/canelaEnRama>  
canela
18. Usa svn info para tener claro que has hecho una copia de trabajo de la rama y pon una captura con el resultado.
  1. Cd canela
  2. svn info

```
estudiante@Estefana:~/canela$ svn info
Path: .
Working Copy Root Path: /home/estudiante/canela
URL: https://svn.code.sf.net/p/examensvnene21/code/branches/canela
EnRama
Relative URL: ^/branches/canelaEnRama
Repository Root: https://svn.code.sf.net/p/examensvnene21/code
Repository UUID: 522dcd5d-ae19-442c-9e5d-186632d9add1
Revision: 3
Node Kind: directory
Schedule: normal
Last Changed Author: egarsan176
Last Changed Rev: 3
Last Changed Date: 2021-02-01 13:45:40 +0100 (lun, 01 feb 2021)

estudiante@Estefana:~/canela$
```

19. Realiza una sesión de trabajo en la rama que has creado siguiendo los siguientes pasos:

1. Actualiza tu copia de trabajo.
2. Svn update
3. Modifica el método compareTo de la clase Persona como hicimos antes. Añade algún fichero, por ejemplo un Readme.md en la raíz del proyecto.
  1. Nano src/prueba/Persona.java (no se ha restaurado la versión original que creo que es la que debía aparecer por lo que he borrado el comentario para hacer algún cambio)
  2. touch Readme.md
4. En la rama trunk añade el atributo color de tipo String a la clase Contenedor. Para ello añade debajo de la línea 7 la siguiente línea:  
private String color;
  1. cd ../listaOrdenada/trunk/
  2. nano src/prueba/Contenedor.javaSube los cambios al repositorio.
  3. svn ci -m "añadido el atributo color de tipo String a Contenedor.java"
5. En la rama canelaEnRama consulta el estado de tu copia de trabajo respecto a la última revisión que actualizaste y respecto a la última revisión que haya en el repositorio. Indica las diferencias y pon una captura de cada una de ellas.
  1. cd ../../canela/
  2. svn status // svn status -u

```
estudiante@Estefana:~/canela$ svn status
?      Readme.md
M      src/prueba/Persona.java
estudiante@Estefana:~/canela$ svn status -u
M      3      src/prueba/Persona.java
?      Readme.md
Status against revision: 4
estudiante@Estefana:~/canela$
```

6. Revisa los cambios realizados en los ficheros que has tocado respecto a la última revisión que actualizaste y respecto de la última revisión que haya en el repositorio.

1. Svn diff src/prueba/Persona.java

```
estudiante@Estefana:~/canela$ svn diff src/prueba/Persona.java
Index: src/prueba/Persona.java
=====
=
--- src/prueba/Persona.java      (revision 3)
+++ src/prueba/Persona.java      (working copy)
@@ -88,7 +88,6 @@

    @Override
    public int compareTo Persona o {
-        // TODO Auto-generated method stub
        return Integer.compare edad o edad;
    }
estudiante@Estefana:~/canela$
```

7. Incorpora los cambios de la rama trunk a la rama canelaEnRama.

1. svn merge <https://svn.code.sf.net/p/examensvnene21/code/trunk/>

8. Envía los cambios al repositorio creando una nueva revisión (*commit*).

1. svn ci -m "fusionada la rama canelaEnRama a la rama trunk"

20. En la rama trunk modifica el método compareTo para que compare por nombre. Cambia las líneas 89 a 92 por estas:

```
@Override
public int compareTo Persona o {
    return Integer.compare nombre, o.nombre;
```

1. cd ../listaOrdenada/trunk/

2. nano src/prueba/Persona.java

21. Reintegra tu rama privada con el desarrollo principal del proyecto (trunk). El método compareTo de la clase Persona debe quedar comparando por edad.

1. svn merge - - reintegrate https://svn.code.sf.net/p/examensvnene21/code/branches/canelaEnRama

2. Te dice que actualices, por lo que hago un svn update

3. Borro la rama que ya no nos sirve  
svn delete <https://svn.code.sf.net/p/examensvnene21/code/branches/canelaEnRama>

4. Se abre el editor de texto y escribo el mensaje