

(A)

```

1. public abstract class MediaAccessor
2. {
3.     public void createNewAlbum(String albumName)
4.         throws PersistenceMechanismException, InvalidPhotoAlbumNameException {
5.
6.         ...
7.         if (albumName.equals("")) {
8.             throw new InvalidPhotoAlbumNameException();
9.         }
10.
11.         String[] names = getAlbumNames();
12.         for (int i = 0; i < names.length; i++) {
13.             if (names[i].equals(albumName))
14.                 throw new InvalidPhotoAlbumNameException();
15.         }
16.         ...
17.         // Create new album and persist it in the file system
18.         ...
19.     }
20.     ...
21. }

```

Exception propagation

(B)

```

1. public class AlbumController extends AbstractController
2. {
3.     public boolean handleCommand(Command command) {
4.
5.         String label = command.getLabel()
6.         ...
7.
8.         if (label.equals("Save")) {
9.             ...
10.             try {
11.                 password = (PasswordScreen) getCurrentScreen();
12.                 getAlbumData().createNewAlbum(albumName.getLabelName());
13.                 getAlbumData().addPassword(albumName.getLabelName(), password.getPassword());
14.             } catch (PersistenceMechanismException e) {
15.                 ...
16.             } catch (InvalidPhotoAlbumNameException e) {
17.                 Alert alert = new Alert(..., "You have provided an invalid Photo Album name", ...);
18.                 ...
19.             }
20.         }
21.         ...
22.     }
23.     ...
24. }

```



Variable feature Privacy