Java Try with Resources

Try/Catch/Finally

```java
InputStream is = null;
FileOutputStream fos = null;
try {
    is = fileStorage.readFile(fd);
    fos = new FileOutputStream(localFileName);

    int len = 0;
    while ((len = is.read(buf)) > 0) {
        fos.write(buf, 0, len);
    }
} catch (IOException e) {
    throw (new FailedException("Could not get input file.", e));
} finally {
    if (is != null) {
        is.close();
    }

    if (fos != null) {
        fos.close();
    }
}
```

Try/Resources

```java
try (InputStream is = fileStorage.readFile(fd); FileOutputStream fos = new FileOutputStream(localFileName);) {
    int len = 0;
    while ((len = is.read(buf)) > 0) {
        fos.write(buf, 0, len);
    }
} catch (IOException e) {
    throw (new FailedException("Could not get input file.", e));
}
```

Custom closeable

```java
public class MyCustomContentProvider implements Closeable {
    public void readContent() throws IOException {
        System.out.println("reading content");
    }

    @Override
    public void close() throws IOException {
        System.out.println("Close custom content provider");
    }
}

@Test
public void testCustomCloseable() throws Exception {
    try (MyCustomContentProvider provider = new MyCustomContentProvider()) {
        provider.readContent();
    } catch (IOException e) {
        // log exception
    }
}
```

Custom closeable output (1)

reading content

Close custom content provider

Custom closeable with exception

```java
@Test
public void testCustomCloseable() throws Exception {
    try (MyCustomContentProvider provider = new MyCustomContentProvider()) {
        throw new IOException();
    } catch (IOException e) {
        // log exception
    }
}
```

Custom closeable output (2)

Close custom content provider