**Objectives**: By the end of this practical exercise, the students should be able to:

* Call the API functions to encrypt / decrypt data using AES
* Investigate the key and data size of AES

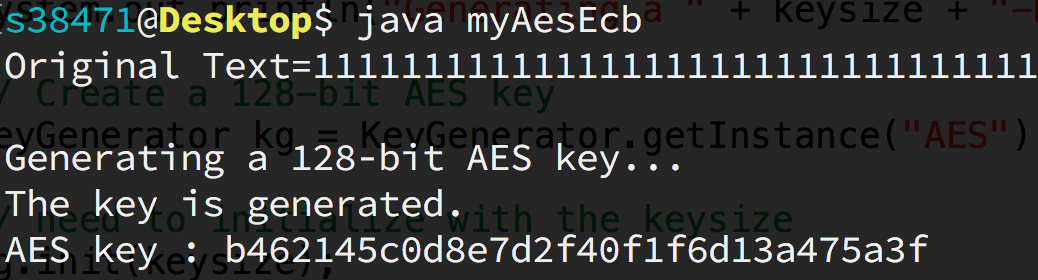
1. Download **myAesEcb,java** from Blackboard. This program is based on **myDesEcb.java** with changes replacing all occurrences of “DES” to “AES” and adjusting the keysize from 56 to 128.
   * To increase clarity in the program, a variable **keysize** is created.
2. Compile and run myAesEcb.java using the following commands:

* **javac myAesEcb.java**
* **java myAesEcb**

1. Run **myAesEcb.class** at least twice, is the ciphertext **identical**? (Note: the plaintext is hardcoded).

**The ciphertext is not identical.**

1. Modify the content of the plaintext to “11……111” (50 x “1”). Compile **myAesEcb.java** and run the modified program.
2. Modify the program to print out the generated AES key (as a string) as shown



1. The AES-128 ciphertext pattern will repeat itself after \_\_\_\_ (bytes). [Given the plaintext is consistent]
2. Modify the initial key size from 128 to 256 as follows:

From [ kg.init(**128**); ] to [ kg.init(**256**); ]

1. Compile and run the modified **myAesEcb.java**.

( Note: If you see the following runtime error ‘*Exception in thread "main" java.security.InvalidKeyException:* ***Illegal key size*** *or default parameter*s’, please **enable unlimited JCE** 1.X.X **cryptography policy** as describe in the last paragraph of practical 1)

1. The AES-256 ciphertext pattern will repeat itself after \_\_\_\_ (bytes). [Given the plaintext is consistent]