

[在此鍵入]

```
#####
```

```
# 安裝 JDK
```

```
#####
```

```
#使用 FireFox 到官方網站下載
```

```
#解壓縮到/opt 底下
```

```
cd ~
```

```
cd Downloads
```

```
tar zxvf jdk-8u121-linux-x64.tar.gz
```

```
sudo mv jdk1.8.0_121 /opt/jdk
```

```
#設定環境變數
```

```
sudo gedit /etc/profile
```

將以下內容複製貼上並保存

```
export
```

```
PATH=$PATH:"/opt/jdk/bin:/opt/hadoop/bin:/opt/hadoop/sbin:/opt/maven/bin:/opt  
/mahout/bin:/opt/eclipse" export CLASS_PATH="/opt/jdk/lib" export  
JAVA_HOME="/opt/jdk" export HADOOP_HOME="/opt/hadoop" export  
MAVEN_HOME="/opt/maven" export MAHOUT_HOME="/opt/mahout"
```

```
#執行環境變數設定
```

```
source /etc/profile
```

```
#測試 jdk
```

```
java -version
```

```
#####
```

```
# 安裝 maven
```

```
#####
```

```
#安裝 maven
```

```
sudo apt install maven
```

[在此鍵入]

#測試 maven

mvn -version

#####

安裝 jclouds

#####

(參考網頁 <https://jclouds.apache.org/guides/openstack/>)

cd ~

mkdir jclouds

cd jclouds

#在 jclouds 目錄下新建一個 pom.xml 文檔,並把以下內容複製貼上並保存
sudo gedit pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <properties>
    <jclouds.version>2.0.1</jclouds.version>
  </properties>
  <groupId>org.apache.jclouds.examples</groupId>
  <artifactId>openstack-examples</artifactId>
  <version>1.0</version>
  <dependencies>
    <!-- jclouds dependencies -->
    <dependency>
      <groupId>org.apache.jclouds.driver</groupId>
      <artifactId>jclouds-slf4j</artifactId>
      <version>${jclouds.version}</version>
    </dependency>
    <dependency>
      <groupId>org.apache.jclouds.driver</groupId>
      <artifactId>jclouds-sshj</artifactId>
```

```
<version>${jclouds.version}</version>
</dependency>
<!-- jclouds OpenStack dependencies -->
<dependency>
  <groupId>org.apache.jclouds.api</groupId>
  <artifactId>openstack-keystone</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.api</groupId>
  <artifactId>openstack-nova</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.api</groupId>
  <artifactId>openstack-swift</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.api</groupId>
  <artifactId>openstack-cinder</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.api</groupId>
  <artifactId>openstack-trove</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.labs</groupId>
  <artifactId>openstack-glance</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.jclouds.labs</groupId>
  <artifactId>openstack-marconi</artifactId>
  <version>${jclouds.version}</version>
</dependency>
```

[在此鍵入]

```
<dependency>
  <groupId>org.apache.jclouds.labs</groupId>
  <artifactId>openstack-neutron</artifactId>
  <version>${jclouds.version}</version>
</dependency>
<!-- 3rd party dependencies -->
<dependency>
  <groupId>ch.qos.logback</groupId>
  <artifactId>logback-classic</artifactId>
  <version>1.0.13</version>
</dependency>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.25</version>
</dependency>
</dependencies>
</project>
```

#配置

```
mvn dependency:copy-dependencies "-DoutputDirectory=./lib"
```

#現在 jclouds 的目錄下會有這些內容

- jclouds/
 - pom.xml
 - lib/
 - *.jar

```
#####
```

```
# 使用程式獲取 Image(鏡像檔)的 ID
```

```
#####
```

```
#創建 java 文檔
```

```
cd ~
```

```
cd jclouds
```

```
sudo gedit JCloudsNova.java
```

```
#把以下內容複製貼上，修改紅字部分，保存
```

[在此鍵入]

```
import com.google.common.collect.ImmutableSet;
import com.google.common.io.Closeables;
import com.google.inject.Module;
import org.jclouds.ContextBuilder;
import org.jclouds.logging.slf4j.config.SLF4JLoggingModule;
import org.jclouds.openstack.nova.v2_0.NovaApi;
import org.jclouds.openstack.nova.v2_0.domain.Image;
import org.jclouds.openstack.nova.v2_0.domain.Flavor;
import org.jclouds.openstack.nova.v2_0.domain.Server;
import org.jclouds.openstack.nova.v2_0.features.ServerApi;
import org.jclouds.openstack.nova.v2_0.features.ImageApi;
import org.jclouds.openstack.nova.v2_0.features.FlavorApi;

import java.io.Closeable;
import java.io.IOException;
import java.util.Set;

public class JCloudsNova implements Closeable {
    private final NovaApi novaApi;
    private final Set<String> regions;

    public static void main(String[] args) throws IOException {
        JCloudsNova jcloudsNova = new JCloudsNova();

        try {
            jcloudsNova.listImages();
            jcloudsNova.close();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            jcloudsNova.close();
        }
    }

    public JCloudsNova() {
        Iterable<Module> modules = ImmutableSet.<Module>of(new
SLF4JLoggingModule());
```

[在此鍵入]

```
String provider = "openstack-nova";
String identity = "admin:admin"; // tenantName:userName
String credential = "你的密碼";

novaApi = ContextBuilder.newBuilder(provider)
    .endpoint("http://你的 IP/identity/v2.0/")
    .credentials(identity, credential)
    .modules(modules)
    .buildApi(NovaApi.class);
regions = novaApi.getConfiguredRegions();
}

private void listImages() {
    for (String region : regions) {
        ImageApi imageApi = novaApi.getImageApi(region);
        System.out.println("Images in " + region);
        for (Image image : imageApi.listInDetail().concat()) {
            System.out.println(" " + image);
        }
    }
}

public void close() throws IOException {
    Closeables.close(novaApi, true);
}
}
```

#編譯

```
javac -classpath ".:lib/*" JCloudsNova.java
```

#執行

```
java -classpath ".:lib/*" JCloudsNova
```

#隨意記錄下一個 Image 的 ID，之後會使用到（用記事本之類的把 ID 保存下來）

[在此鍵入]

```
islab@islab-virtual-machine:~/jclouds
:14:03Z", "minDisk": 0, "progress": 100, "minRam": 0, "metadata": {}}]]]"
Image{id=d7c8e68e-9447-49f0-8189-8acec016ef6f, name=cirros-0.3.4-x86_64-uec, li
nks=[Link{relation=SELF, href=http://192.168.47.133:8774/v2.1/images/d7c8e68e-94
47-49f0-8189-8acec016ef6f}, Link{relation=BOOKMARK, href=http://192.168.47.133:8
774/images/d7c8e68e-9447-49f0-8189-8acec016ef6f}, Link{relation=ALTERNATE, type=
application/vnd.openstack.image, href=http://192.168.47.133:9292/images/d7c8e68e
-9447-49f0-8189-8acec016ef6f}], updated=Wed Dec 21 18:14:10 CST 2016, created=We
d Dec 21 18:14:09 CST 2016, tenantId=null, userId=null, status=ACTIVE, progress=
100, minDisk=0, minRam=0, server=null, metadata={kernel_id=4824add0-3985-4c17-ad
29-94ce21f5d357, ramdisk_id=e9d68983-9a35-444d-865e-f2f0dc9efbf8}}
Image{id=e9d68983-9a35-444d-865e-f2f0dc9efbf8, name=cirros-0.3.4-x86_64-uec-ram
disk, links=[Link{relation=SELF, href=http://192.168.47.133:8774/v2.1/images/e9d
68983-9a35-444d-865e-f2f0dc9efbf8}, Link{relation=BOOKMARK, href=http://192.168.
47.133:8774/images/e9d68983-9a35-444d-865e-f2f0dc9efbf8}, Link{relation=ALTERNAT
E, type=application/vnd.openstack.image, href=http://192.168.47.133:9292/images/
e9d68983-9a35-444d-865e-f2f0dc9efbf8}], updated=Wed Dec 21 18:14:06 CST 2016, cr
eated=Wed Dec 21 18:14:06 CST 2016, tenantId=null, userId=null, status=ACTIVE, p
rogress=100, minDisk=0, minRam=0, server=null, metadata={}}
Image{id=4824add0-3985-4c17-ad29-94ce21f5d357, name=cirros-0.3.4-x86_64-uec-ker
nel, links=[Link{relation=SELF, href=http://192.168.47.133:8774/v2.1/images/4824
add0-3985-4c17-ad29-94ce21f5d357}, Link{relation=BOOKMARK, href=http://192.168.4
7.133:8774/images/4824add0-3985-4c17-ad29-94ce21f5d357}, Link{relation=ALTERNATE
, type=application/vnd.openstack.image, href=http://192.168.47.133:9292/images/4
824add0-3985-4c17-ad29-94ce21f5d357}], updated=Wed Dec 21 18:14:04 CST 2016, cre
```

#####

使用程式獲取 Flavor(cpu 配置、內存配置等)的 ID

#####

#回到 jclouds 目錄編輯 java 文檔

sudo gedit JCloudsNova.java

#把以下內容複製貼上，修改紅字部分，保存

```
import com.google.common.collect.ImmutableSet;
import com.google.common.io.Closeables;
import com.google.inject.Module;
import org.jclouds.ContextBuilder;
import org.jclouds.logging.slf4j.config.SLF4JLoggingModule;
import org.jclouds.openstack.nova.v2_0.NovaApi;
import org.jclouds.openstack.nova.v2_0.domain.Image;
import org.jclouds.openstack.nova.v2_0.domain.Flavor;
import org.jclouds.openstack.nova.v2_0.domain.Server;
import org.jclouds.openstack.nova.v2_0.features.ServerApi;
import org.jclouds.openstack.nova.v2_0.features.ImageApi;
import org.jclouds.openstack.nova.v2_0.features.FlavorApi;

import java.io.Closeable;
import java.io.IOException;
```

[在此鍵入]

```
import java.util.Set;

public class JCloudsNova implements Closeable {
    private final NovaApi novaApi;
    private final Set<String> regions;

    public static void main(String[] args) throws IOException {
        JCloudsNova jcloudsNova = new JCloudsNova();

        try {
            jcloudsNova.listFlavors();
            jcloudsNova.close();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            jcloudsNova.close();
        }
    }

    public JCloudsNova() {
        Iterable<Module> modules = ImmutableSet.<Module>of(new
SLF4JLoggingModule());

        String provider = "openstack-nova";
        String identity = "admin:admin"; // tenantName:userName
        String credential = "你的密碼";

        novaApi = ContextBuilder.newBuilder(provider)
            .endpoint("http://你的 IP/identity/v2.0/")
            .credentials(identity, credential)
            .modules(modules)
            .buildApi(NovaApi.class);
        regions = novaApi.getConfiguredRegions();
    }

    private void listFlavors() {
        for (String region : regions) {
            FlavorApi flavorApi = novaApi.getFlavorApi(region);
        }
    }
}
```


[在此鍵入]

```
        System.out.println("flavor in " + region);
        for (Flavor flavor : flavorApi.listInDetail().concat()) {
            System.out.println(" " + flavor);
        }
    }

    public void close() throws IOException {
        Closeables.close(novaApi, true);
    }
}
```

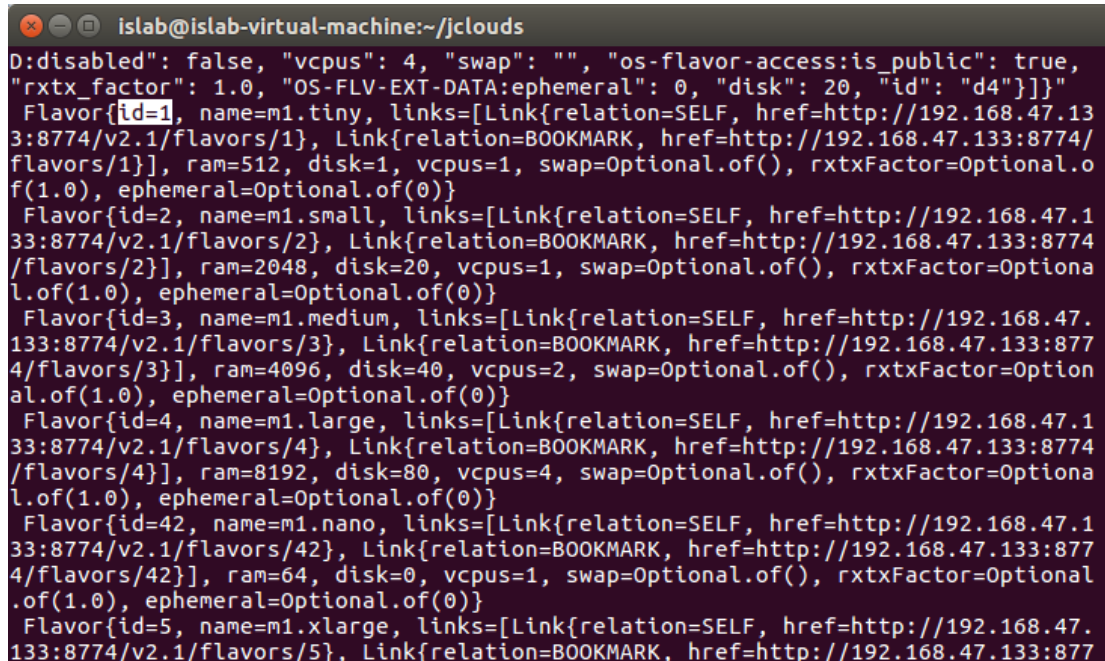
#編譯

```
javac -classpath ".:lib/*" JCloudsNova.java
```

#執行

```
java -classpath ".:lib/*" JCloudsNova
```

#記錄下一個 Flavor 的 ID，之後會使用到（用記事本之類的把 ID 保存下來）



```
islab@islab-virtual-machine:~/jclouds
D:disabled": false, "vcpus": 4, "swap": "", "os-flavor-access:is_public": true,
"rxtx_factor": 1.0, "OS-FLV-EXT-DATA:ephemeral": 0, "disk": 20, "id": "d4"]]]
  Flavor{id=1, name=m1.tiny, links=[Link{relation=SELF, href=http://192.168.47.13
3:8774/v2.1/flavors/1}, Link{relation=BOOKMARK, href=http://192.168.47.133:8774/
flavors/1}], ram=512, disk=1, vcpus=1, swap=Optional.of(), rxtxFactor=Optional.o
f(1.0), ephemeral=Optional.of(0)}
  Flavor{id=2, name=m1.small, links=[Link{relation=SELF, href=http://192.168.47.1
33:8774/v2.1/flavors/2}, Link{relation=BOOKMARK, href=http://192.168.47.133:8774
/flavors/2}], ram=2048, disk=20, vcpus=1, swap=Optional.of(), rxtxFactor=Optiona
l.of(1.0), ephemeral=Optional.of(0)}
  Flavor{id=3, name=m1.medium, links=[Link{relation=SELF, href=http://192.168.47.
133:8774/v2.1/flavors/3}, Link{relation=BOOKMARK, href=http://192.168.47.133:877
4/flavors/3}], ram=4096, disk=40, vcpus=2, swap=Optional.of(), rxtxFactor=Option
al.of(1.0), ephemeral=Optional.of(0)}
  Flavor{id=4, name=m1.large, links=[Link{relation=SELF, href=http://192.168.47.1
33:8774/v2.1/flavors/4}, Link{relation=BOOKMARK, href=http://192.168.47.133:8774
/flavors/4}], ram=8192, disk=80, vcpus=4, swap=Optional.of(), rxtxFactor=Optiona
l.of(1.0), ephemeral=Optional.of(0)}
  Flavor{id=42, name=m1.nano, links=[Link{relation=SELF, href=http://192.168.47.1
33:8774/v2.1/flavors/42}, Link{relation=BOOKMARK, href=http://192.168.47.133:877
4/flavors/42}], ram=64, disk=0, vcpus=1, swap=Optional.of(), rxtxFactor=Optional
.of(1.0), ephemeral=Optional.of(0)}
  Flavor{id=5, name=m1.xlarge, links=[Link{relation=SELF, href=http://192.168.47.
133:8774/v2.1/flavors/5}, Link{relation=BOOKMARK, href=http://192.168.47.133:877
```

[在此鍵入]

```
#####
```

```
# 使用程式新增虛擬機(instance)
```

```
#####
```

```
# 回到 jclouds 目錄編輯 java 文檔
```

```
sudo gedit JCloudsNova.java
```

#把以下內容複製貼上，修改紅字部分，保存

```
import com.google.common.base.Optional;
import org.jclouds.ContextBuilder;
import org.jclouds.net.domain.IpProtocol;
import org.jclouds.openstack.nova.v2_0.NovaApi;
import org.jclouds.openstack.nova.v2_0.domain.*;
import org.jclouds.openstack.nova.v2_0.extensions.FloatingIPApi;
import org.jclouds.openstack.nova.v2_0.extensions.KeyPairApi;
import org.jclouds.openstack.nova.v2_0.extensions.SecurityGroupApi;
import org.jclouds.openstack.nova.v2_0.features.FlavorApi;
import org.jclouds.openstack.nova.v2_0.features.ImageApi;
import org.jclouds.openstack.nova.v2_0.features.ServerApi;
import org.jclouds.openstack.nova.v2_0.options.CreateServerOptions;
import org.jclouds.openstack.nova.v2_0.predicates.ServerPredicates;

import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.nio.file.attribute.PosixFilePermission;
import java.util.HashSet;
import java.util.List;
import java.util.Set;
import java.util.concurrent.TimeUnit;
import java.util.stream.Collectors;

import static java.lang.System.out;

class JCloudsNova {

    public static void main(String[] args) throws IOException {
```

[在此鍵入]

```
        out.println("=====");
        String provider = "openstack-nova";
        String identity = "admin:admin";
        String credential = "你的密碼";
        String authUrl = "http://你的 IP/identity/v2.0/";

        NovaApi conn = ContextBuilder.newBuilder(provider)
                .endpoint(authUrl)
                .credentials(identity, credential)
                .buildApi(NovaApi.class);
        String region = conn.getConfiguredRegions().iterator().next();
        out.println("Running in region: " + region);

        String imageId = "4824add0-3985-4c17-ad29-94ce21f5d357(之前獲取的 Image ID)";

        Image retrievedImage = conn.getImageApi(region).get(imageId);
        out.println(retrievedImage.toString());

        String flavorId = "1(之前獲取的 Flavor ID)";
        Flavor flavor = conn.getFlavorApi(region).get(flavorId);
        out.println(flavor.toString());

        String testingInstance = "testingInstance(instance 的名稱)";
        ServerCreated testInstance =
conn.getServerApi(region).create(testingInstance, imageId, flavorId);
        out.println("Server created. ID: " + testInstance.getId());
    }
}
```

#編譯

```
javac -classpath ".:lib/*" JCloudsNova.java
```

#執行

```
java -classpath ".:lib/*" JCloudsNova
```

#執行結果如下，並記錄下此 ID，稍後刪除 instance 會用到

```
islab@islab-virtual-machine:~/jclouds$
Running in region: RegionOne
Images in region:
  cirros-0.3.4-x86_64-uec
  cirros-0.3.4-x86_64-uec-randisk
  cirros-0.3.4-x86_64-uec-kernel
islab@islab-virtual-machine:~/jclouds$ javac -classpath ".:lib/*" GettingStarted.java
islab@islab-virtual-machine:~/jclouds$ java -classpath ".:lib/*" GettingStarted
=====
Running in region: RegionOne
Image{id=4824add0-3985-4c17-ad29-94ce21f5d357, name=cirros-0.3.4-x86_64-uec-kernel, links=[Link{relation=SELF, href=http://192.168.47.133:8774/v2.1/images/4824add0-3985-4c17-ad29-94ce21f5d357}, Link{relation=BOOKMARK, href=http://192.168.47.133:8774/images/4824add0-3985-4c17-ad29-94ce21f5d357}, Link{relation=ALTERNATE, type=application/vnd.openstack.image, href=http://192.168.47.133:9292/images/4824add0-3985-4c17-ad29-94ce21f5d357}], updated=Wed Dec 21 18:14:04 CST 2016, created=Wed Dec 21 18:14:03 CST 2016, tenantId=null, userId=null, status=ACTIVE, progress=100, minDisk=0, minRam=0, server=null, metadata={}}
Flavor{id=1, name=m1.tiny, links=[Link{relation=SELF, href=http://192.168.47.133:8774/v2.1/flavors/1}, Link{relation=BOOKMARK, href=http://192.168.47.133:8774/flavors/1}], ram=512, disk=1, vcpus=1, swap=Optional.of(), rxtxFactor=Optional.of(1.0), ephemeral=Optional.of(0)}
Server created. ID: 5e21ea6b-0444-42be-8472-180c211acb65
islab@islab-virtual-machine:~/jclouds$
```

The screenshot shows the OpenStack dashboard interface. The top navigation bar includes the OpenStack logo, a search bar, and a user profile dropdown labeled 'admin'. The left sidebar contains a menu with items like Project, Compute, Overview, Instances (highlighted in blue), Volumes, Images, Access & Security, Network, Admin, and Identity. The main content area is titled 'Instances' and displays a table of running instances. A red box highlights the 'admin' user profile in the top right corner of the dashboard.

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Action
testingInstance	cirros-0.3.4-x86_64-uec-kernel	<ul style="list-style-type: none"> 172.24.4.12 2001:db8::4 	m1.tiny	-	Active	nova	None	Running	7 minutes	Create

12

[在此鍵入]

回到 jclouds 目錄編輯 java 文檔

sudo gedit JCloudsNova.java

#把以下內容複製貼上，修改紅字部分，保存

```
import com.google.common.base.Optional;
import org.jclouds.ContextBuilder;
import org.jclouds.net.domain.IpProtocol;
import org.jclouds.openstack.nova.v2_0.NovaApi;
import org.jclouds.openstack.nova.v2_0.domain.*;
import org.jclouds.openstack.nova.v2_0.extensions.FloatingIPApi;
import org.jclouds.openstack.nova.v2_0.extensions.KeyPairApi;
import org.jclouds.openstack.nova.v2_0.extensions.SecurityGroupApi;
import org.jclouds.openstack.nova.v2_0.features.FlavorApi;
import org.jclouds.openstack.nova.v2_0.features.ImageApi;
import org.jclouds.openstack.nova.v2_0.features.ServerApi;
import org.jclouds.openstack.nova.v2_0.options.CreateServerOptions;
import org.jclouds.openstack.nova.v2_0.predicates.ServerPredicates;

import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.nio.file.attribute.PosixFilePermission;
import java.util.HashSet;
import java.util.List;
import java.util.Set;
import java.util.concurrent.TimeUnit;
import java.util.stream.Collectors;

import static java.lang.System.out;

class JCloudsNova {

    public static void main(String[] args) throws IOException {
        out.println("=====");
        String provider = "openstack-nova";
        String identity = "admin:admin";
```

[在此鍵入]

```
String credential = "你的密碼";
String authUrl = "http://你的 IP/identity/v2.0/";

NovaApi conn = ContextBuilder.newBuilder(provider)
    .endpoint(authUrl)
    .credentials(identity, credential)
    .buildApi(NovaApi.class);
String region = conn.getConfiguredRegions().iterator().next();
out.println("Running in region: " + region);

String instanceId = "5e21ea6b-0444.....(之前獲取的 instance ID)";
ServerApi serverApi = conn.getServerApi(region);

if (serverApi.delete(instanceId)) {
    out.println("Server being deleted, please wait.");
    ServerPredicates.awaitStatus(serverApi, Server.Status.DELETED, 600,
5).apply(instanceId);
    serverApi.list().concat().forEach(instance -> out.println(" " +
instance));
} else {
    out.println("Server not deleted.");
}
}
```

#編譯

```
javac -classpath ".:lib/*" JCloudsNova.java
```

#執行

```
java -classpath ".:lib/*" JCloudsNova
```

#執行結果如下則成功刪除

```
=====
Running in region: RegionOne
Server being deleted, please wait.
isl@isl-virtual-machine:~/jclouds$
```