



# Jar files

Package your classes in one file



# What is a JAR file?

*A JAR (Java ARchive) is a package file format typically used to aggregate many Java class files and associated metadata and resources (text, images, etc.) into one file for distribution. [Wikipedia \(CC BY-SA 3.0\)](#)*

A file with an archive of Java class files (and other files), which can be used by some application.

With a JAR file on the classpath, we can import packages and classes from the archive and use them in an application.

# How to create a JAR file

Simplest way:

```
jar cvf some-file-name.jar some-directory
```

- The cvf directive stands for Create Verbose File
- some-file-name.jar will be created
- some-directory contains the files to be included

# Example

We have a class Student, which we want to package in a JAR file:

```
src/  
└─ se  
    └─ itu  
        └─ students  
            └─ domain  
                └─ Student.java
```

```
package se.itu.students.domain;

public class Student {
    private String name;
    private String email;

    public Student(String name, String email) {
        this.name = name;
        this.email = email;
    }

    public String name() {
        return name;
    }

    public String email() {
        return email;
    }

    @Override
    public String toString() {
        return new StringBuilder("Student with name <")
            .append(name).append("> and email <")
            .append(email).append(">").toString();
    }
}
```

# Compile the file to bin/

```
$ javac -d bin src/se/itu/students/domain/Student.java
```

```
bin/
```

```
└─ se
    └─ itu
        └─ students
            └─ domain
                └─ Student.class
```

# Create the JAR from se/

```
$ cd bin
```

```
$ jar cvf students.jar se
```

```
added manifest
```

```
adding: se/(in = 0) (out= 0) (stored 0%)
```

```
adding: se/itu/(in = 0) (out= 0) (stored 0%)
```

```
adding: se/itu/students/(in = 0) (out= 0) (stored 0%)
```

```
adding: se/itu/students/domain/(in = 0) (out= 0) (stored 0%)
```

```
adding: se/itu/students/domain/Student.class(in = 745) (out=
424) (deflated 43%)
```

# Application using the JAR

```
test/  
└─ TestStudents.java
```

```
# Uses se.itu.students.domain.Student  
# But where does it get it from?
```



# Application using the JAR

```
package test;

import se.itu.students.domain.Student;

public class TestStudents {
    public static void main(String[] args) {
        Student[] students = {
            new Student("Henrik", "henke@gu.se"),
            new Student("Rikard", "rille@gu.se"),
            new Student("Aida", "aida@gu.se"),
            new Student("Urban", "ubbe@gu.se")
        };
        for (Student stud : students) {
            System.out.println(stud);
        }
    }
}
```

# Application using the JAR

```
$ javac -cp .:students.jar test/TestStudents.java
```

```
# On windows: javac -cp .;students.jar test/TestStudents.java (semicolon)
```

```
$ java -cp .:students.jar test.TestStudents
```

```
# Both compilation and running requires the JAR on the classpath!
```

# Further reading

<https://docs.oracle.com/javase/tutorial/deployment/jar/basicsindex.html>

<https://docs.oracle.com/javase/8/docs/technotes/guides/jar/jarGuide.html>

[https://en.wikipedia.org/wiki/JAR\\_\(file\\_format\)](https://en.wikipedia.org/wiki/JAR_(file_format))