

matlabcontrol allows us to use MATLAB in our Java code. All we need is

- the jar file matlabcontrol-4.1.0.jar, and
- its API.

Using MATLAB in Java

- Connect with MATLAB;
- Convert data from Java to MATLAB;
- Execute MATLAB code;
- Convert data from MATLAB to Java.

Connect with MATLAB

```
import matlabcontrol.MatlabProxyFactory;
import matlabcontrol.MatlabProxy;
import matlabcontrol.MatlabConnectionException;
import matlabcontrol.MatlabInvocationException;

public class Test
{
    public static void main(String[] args) throws
        MatlabConnectionException, MatlabInvocationException
    {
        MatlabProxyFactory factory = new MatlabProxyFactory();
        MatlabProxy proxy = factory.getProxy();

        // use MATLAB

        proxy.disconnect();
    }
}
```

Execute MATLAB code

```
proxy.eval("fprintf('hello world\\n');");
```

Convert data from Java to MATLAB: simple

```
proxy.setVariable("birthYear", 1995);
```

Execute MATLAB code

```
proxy.eval("now = clock;");  
proxy.eval("currentYear = now(1);");  
proxy.eval("age = currentYear - birthYear;");
```

Convert data from MATLAB to Java: simple

```
Object result = proxy.getVariable("age");
```

Convert data from MATLAB to Java: simple

```
double[] result = (double[]) proxy.getVariable("age");  
double age = result[0];  
output.printf("%.0f\n", age);
```


Convert data from Java to MATLAB: arrays

```
double[][] matrix = {{ 1, 2, 3 },
                     { 4, 5, 6 },
                     { 7, 8, 9 }};
MatlabTypeConverter converter =
    new MatlabTypeConverter(proxy);
MatlabNumericArray array =
    new MatlabNumericArray(matrix, null);
converter.setNumericArray("matrix", array);
```

Convert data from Java to MATLAB: arrays

```
double[] [] matrix = {{ 1, 2, 3 },
                       { 4, 5, 6 },
                       { 7, 8, 9 }};
MatlabTypeConverter converter =
    new MatlabTypeConverter(proxy);
MatlabNumericArray array =
    new MatlabNumericArray(matrix, null);
converter.setNumericArray("matrix", array);
proxy.eval("matrix = transpose(matrix);");
```

Convert data from MATLAB to Java: arrays

```
matrix =
    converter.getNumericArray("matrix").getRealArray2D();

for (int r = 0; r < matrix.length; r++)
{
    for (int c = 0; c < matrix[r].length; c++)
    {
        output.printf("%.0f ", matrix[r][c]);
    }
    output.println();
}
```

Some MATLAB commands:

- `javaclasspath`: returns MATLAB's class path
- `javaaddpath`: add to MATLAB's class path
`javaaddpath({'./type.jar'});`
- `javarmpath`: remove from MATLAB's class path

Using Java in MATLAB

Invoking a static method

```
pi = java.lang.Double.parseDouble('3.14');
```

Note that MATLAB converts the character array '3.14' to a String.

Creating an object

```
first = type.lib.Fraction(1, 2);
```

Invoking non-static methods

```
first = type.lib.Fraction(1, 2);  
second = type.lib.Fraction(1, 2);  
add(first, second);  
toString(first)
```

Question

- Let the user choose a csv file with energy use.
- Let user select a time frame w from a drop down with values 1–10.
- Show a graph of Ontario's hourly energy use during the last w weeks.