

Adding a mobile

```
private void addMobileActionHandler() {
    Check if model is empty
    if (this.modelTextField.getText().isEmpty()) {
        showMessage("Invalid Model!")
    }
    Check if price is empty or not a valid number
    else if (this.priceTextField.getText().isEmpty() || !isValidNum(this.priceTextField.getText())) {
        showMessage("Invalid Price!")
    }
    Check if weight is empty or not a valid number
    else if (this.weightTextField.getText().isEmpty() || !isValidNum(this.weightTextField.getText())) {
        showMessage("Invalid Weight!")
    }
    Check if size is empty
    else if (this.sizeTextField.getText().isEmpty()) {
        showMessage("Invalid Size!")
    }
    Check if initial credit is empty or not a valid number
    else if (this.initialCreditTextField.getText().isEmpty() ||
!isValidNum(this.initialCreditTextField.getText())) {
        showMessage("Invalid Credit!")
    }
    If all values are valid, add a new mobile to the list
    else {
        this.gadgetList.add(
```

Adding a Mp3

```
private void addMP3ActionHandler() {
    if (this.modelTextField.getText().isEmpty()) {
        showMessage("Invalid Model!")
    }
    else if (this.priceTextField.getText().isEmpty() || !isValidNum(this.priceTextField.getText())) {
        showMessage("Invalid Price!")
    }
    else if (this.weightTextField.getText().isEmpty() || !isValidNum(this.weightTextField.getText())) {
        showMessage("Invalid Weight!")
    }
    else if (this.sizeTextField.getText().isEmpty()) {
        showMessage("Invalid Size!")
    }
    else if (this.initialMemoryTextField.getText().isEmpty() ||
!isValidNum(this.initialMemoryTextField.getText())) {
```

```

        showErrorMessage("Invalid Memory!")
    }
    else {
        We reach here only if all values are valid. So a MP3 is added.
        this.gadgetList.add(new Mp3(
            modelTextField.getText(),
            Double.parseDouble(priceTextField.getText()),
            Integer.parseInt(weightTextField.getText()),
            sizeTextField.getText(),
            Integer

```

Displaying all the gadgets in a array list

```

public class GadgetDisplay {

    private List<Gadget> gadgetList;

    public GadgetDisplay(List<Gadget> gadgetList) {
        this.gadgetList = gadgetList;
    }

    private void displayAllActionListener() {

        String[] columnNames = {"Model", "Price", "Weight", "Size", "Credit", "Memory"};
        Object[][] data = new Object[gadgetList.size()][columnNames.length];

        for (int i = 0; i < this.gadgetList.size(); i++) {
            Gadget gadget = gadgetList.get(i);

            data[i][0] = gadget.getModel();
            data[i][1] = gadget.getPrice();
            data[i][2] = gadget.getWeight();
            data[i][3] = gadget.getSize();

            if (gadget instanceof Mobile) {
                Mobile mob = (Mobile) gadget;
                data[i][4] = mob.getCreditRemaining();
            }
            else {
                data[i][4] = "--";
            }

            if (gadget instanceof Mp3) {
                Mp3 mp3 = (Mp3) gadget;

```

```

        data[i][5] = mp3.getMemoryAvailable();
    }
    else {
        data[i][5] = "--";
    }
}

```

Create a new table model

```
DefaultTableModel model = new DefaultTableModel(data, columnNames);
```

Create a new JTable with the table model

```
JTable table = new JTable(model);
```

Wrap the table in a JScrollPane for scrolling

```
JScrollPane scrollPane = new JScrollPane(table);
```

Create a JFrame and add the scroll pane

```
JFrame frame = new JFrame("Gadget List");
frame.add(scrollPane, BorderLayout.CENTER);
```

```
frame.setSize(600, 400);
```

```
frame.setVisible(true);
```

```

    }
}

```

Making a call

```

private void makeCallActionListener() {
    if (phoneNumberTextField.getText().isEmpty()) {
        showErrorMessage("Invalid Phone Number!")
    }
    else if (durationTextField.getText().isEmpty() || !isValidNum(durationTextField.getText())) {
        showErrorMessage("Invalid Duration!")
    }
    else if (displayNumberTextField.getText().isEmpty() ||
!isValidNum(displayNumberTextField.getText())) {
        showErrorMessage("Invalid Display Number!")
    }
    else {
        int selectedGadget = Integer.parseInt(displayNumberTextField.getText())

        if (selectedGadget < 0 || selectedGadget >= gadgetList.size()) {
            Falls out of range.
            showErrorMessage("The selected gadget index is out of range!")
        }
    }
}

```

```

    }
    else {
        Gadget gadget = gadgetList.get(selectedGadget)
        if (gadget instanceof Mobile) {
            // It is a Mobile so making the call.
            Mobile mob = (Mobile) gadget
            String response = mob.call(phoneNumberTextField.getText(),
Integer.parseInt(durationTextField.getText()))

            showInformationMessage(response)
        }
        else {
            showErrorMessage("The selected gadget is an MP3. It doesn't support calls.")
        }
    }
}
}

```

Downloading Music

```

private void downloadMusicActionListener() {
    if (downloadSizeTextField.getText().isEmpty() ||
!isValidNum(downloadSizeTextField.getText())) {
        showErrorMessage("Invalid Download Size!")
    }
    else if (displayNumberTextField.getText().isEmpty() ||
!isValidNum(displayNumberTextField.getText())) {
        showErrorMessage("Invalid Display Number!")
    }
    else {
        selectedGadget = Integer.parseInt(displayNumberTextField.getText())

        if (selectedGadget < 0 || selectedGadget >= gadgetList.size()) {
            Falls out of range.
            showErrorMessage("The selected gadget index is out of range!")
        }
        else {
            gadget = gadgetList.get(selectedGadget)
            if (gadget instanceof Mp3) {
                // It is a Mobile so making the call.
                mp3 = (Mp3) gadget
                response = mp3.downloadMusic(Integer.parseInt(downloadSizeTextField.getText()))

                showInformationMessage(response)
            }
        }
    }
}

```

```
    else {  
        showErrorMessage("The selected gadget is a Mobile. It doesn't support downloads.")  
    }  
}  
}
```