

# Basic UI Set Up

This section provides information in setting up a basic UI for the UIP API. This type of set up is exceedingly simple and requires very little code. In short, a plug-in may define its UI as a list of `Parameter` objects which define how to display and persist the required user input data.

The three below methods plus some understanding of the `Parameter` class and `ViewOptions` are all that is required to create a simple static UI.

- `protected void setupParameters()`
- `protected final void addParameter(Parameter p)`
- `public final Object getParameterValue(String key)`
- `public void setParameterValue(String key, Object value)`
- `public void clearParameterValue(String key)`
- `public void getParameter(String key)`

## `protected void setupParameters()`

Plug-ins should implement this method to provide the API with the UI definitions required for configuration. Generally, the plug-in will create `Parameter` objects and add them by calling `addParameter(Parameter)`.

Example:

```
@Override
protected void setupParameters() {
    Parameter p = new Parameter();
    ...

    addParameter(p);
}
```

[top](#)

---

## `protected final void addParameter(Parameter p)`

Plug-ins should use this method to notify the API of any new parameters that it creates. Note that new parameters will not be persisted if they are added outside of the `setupParameters` method. See above example.

[top](#)

---

## `public final Object getParameterValue(String key)`

This method returns the current value for the parameter with unique key 'key'.

Example:

```
Object paramVal = getParameterValue("SOME_KEY");
if (paramVal != null && paramVal.equals("INTERESTING_CONFIG")) {
    // Do something interesting
    ...
}
```

[top](#)

---

## **public void setParameterValue(String key, Object value)**

This method sets the stored value for a parameter key.

Example:

```
Object paramVal = getParameterValue("SOME_KEY");
if (shouldChangeThisValue(paramVal)) {
    Object newValue = new Object();
    setParameterValue("SOME_KEY", newValue);
}
```

[top](#)

---

## **public void clearParameterValue(String key)**

Equivalent to calling `setParameterValue(key, null)`; except that this method also removes the parameter key from the parameter values map.

Example:

```
Object paramVal = getParameterValue("SOME_KEY");
if (shouldRemoveThisValue(paramVal)) {
    clearParameterValue("SOME_KEY");
}
```

[top](#)

---

## **public void getParameter(String key)**

Retrieves a previously set up parameter with the given key. Returns null if no parameter with this key exists.

Example:

```
if (needToUpdateParameter()) {  
    Parameter p = getParameter("SOME_KEY");  
    ...  
}
```

[top](#)

**Previous topic:** [Parameter class](#)

**Next topic:** [Dynamic UI set up](#)