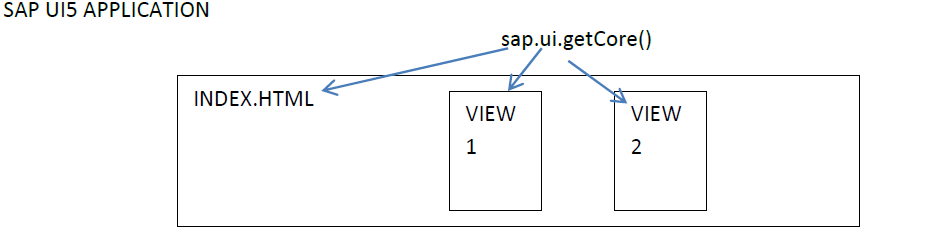
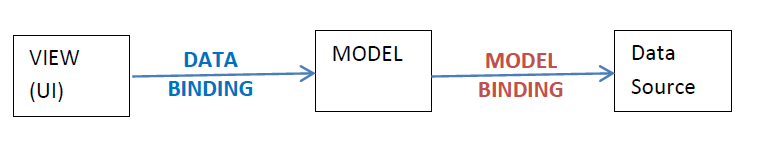
Difference between sap.ui.getCore() and this.getView()

this.getView() local parameter which has access to its only view controller

sap.ui.getCore() is global parameter

Example:



**DATA BINDING in UI5**

Binding between **MODELS to DATA SOURCE** is **MODEL BINDING.**

Binding between **USERINTERFACE (VIEW) to MODEL** is called as DATA BINDING

Whenever there is a change in DATASOURCE Model gets changed. When model gets changed UI gets changed. And it happens vice versa as well.

**POINTS:**

Data Source can be from SAP / NON-SAP / JAVA Platform / can be flat file. (sometimes it can be from MODEL, OBJECT also).

**BINDING TYPES:**

* ONE WAY BINDING
* TWO WAY BINDING

One way Binding and Two way Binding are with reference to MODEL. Not with Data Source.

**MODEL TYPES:**

* JSON MODEL: when data is in format of JSON. I use JSON Model
* XML MODEL: when data is in format of XML. I use XML Model.
* ODATA MODEL: when data is in format of ODATA. I use ODATA Model. (Preferred by SAP UI5)
* RESOUCE MODEL: I18N
* **BINDING TYPES** 
  + **ELEMENT BINDING:** Binding to Control when I find entry. (FORM) I have table, when I select particular record. It has to show me in form of SIMPLE FORM or FORM.
* Note: bunch of data we will have inform of ARRAY.
  + **PROPERTY BINDING:** binding is between PROPERTY OF A CONTROL (TEXT OF LABEL) to **PROPERTY OF A MODEL**
* **AGGREGATION BINDING:** when I have collection of data on **table or list or drop down box / combo box.**

When we are working on Path Binding we need to add this line code in Index.html. then only Its working.

**data-sap-ui-xx-bindingSyntax="complex"**

Element Binding

* Forms – collection of data to display
* Address: hno, street, city
* A selected entry from a table / collection.

Element binding allows to bind elements to a specific object in the model data, which will create a binding context and allow relative binding within the control and all of its children. This is especially helpful in master/detail scenarios.

To define an element binding, use the bindElement method on a control:

oControl.bindElement("/company");

oControl.bindProperty("value", "name");