1. qmlRegisterType
2. Step

B1: Define class

B2: qmlRegisterType<Config>("BackEnd", 1, 0, "Config"); in main.cpp

B3: create object in qml

1. Use case
2. Q\_PROPERTY(int price1 READ getPrice WRITE setPrice NOTIFY priceChanged)

* Obejct in qml has property price1
* Obejct has 2 trigger function: *onPriceChanged,* onPrice1Changed

1. If a function is slot => can be called from qml, can not call a function that is not a slot function
2. To call a function that is not a slot => add Q\_INVOKABLE at the prefix of this function
3. setContextProperty
4. Step

B1: Define class

B2: *Config* *config;*

*//* *engine.rootContext()->setContextProperty("setContext",* *&config);*

B3: call in qml