Detecting day of week in QML

This article explains how to detect day of week in QML, how to detect localized name, how to link localized name with Qt enum.The code example utilizes the formatDateTime method.

QML global `Qt` object provides enum for defining days of week:

```
Qt.Monday  = 1
Qt.Tuesday = 2
...
Qt.Sunday  = 7
```

Unfortunately, `Date` object does not contain functionality for using this enum. But we can use `Qt.formatDateTime()` function for detecting localized name of day:

```
var myDate = new Date() // current date
var localizedDayName = Qt.formatDateTime( myDate, "dddd" ) // current day name
```

We can use a little trick for linking localized day name with Qt enum:

```
function localizedDayNameByIdx( idx )
{
    if( idx >= Qt.Monday && idx <= Qt.Sunday )
    {
        var myDate = new Date()

        myDate.setFullYear( 1980, 2, 17 ) // March 17, 1980 - we know, it was monday
        myDate.setDate( myDate.getDate() + idx - 1 ) // Qt.Monday = 1

        return Qt.formatDateTime( myDate, "dddd" )
    }

    return ""
}

function dayIdxByLocalizedDayName( dayName )
{
    var res = -1

    var myDate = new Date()
    myDate.setFullYear( 1980, 2, 17 ) // March 17, 1980 - we know, it was monday
    for( var i = 0; i < 7; i ++)
    if( Qt.formatDateTime( myDate, "dddd" ) == dayName )
    {
        res = i + 1;
        break
    }
    else
        myDate.setDate( myDate.getDate() + 1 )

    return res
}
```

Example of use for the Russian localization:

```
- localizedDayNameByIdx( Qt.Monday ) // returns "понедельник"
- dayIdxByLocalizedDayName( "понедельник" ) // returns Qt.Monday
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.