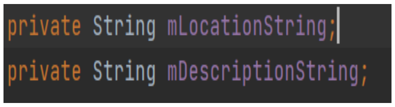
Naming Conventions

Variable Names:

Avoid DisInformation:

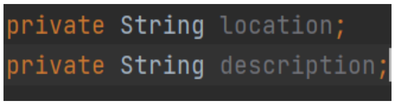
Bad practice:



```
private String mLocationString;
private String mDescriptionString;
```

We should Avoid unnecessary encodings of data types along with the variable name.

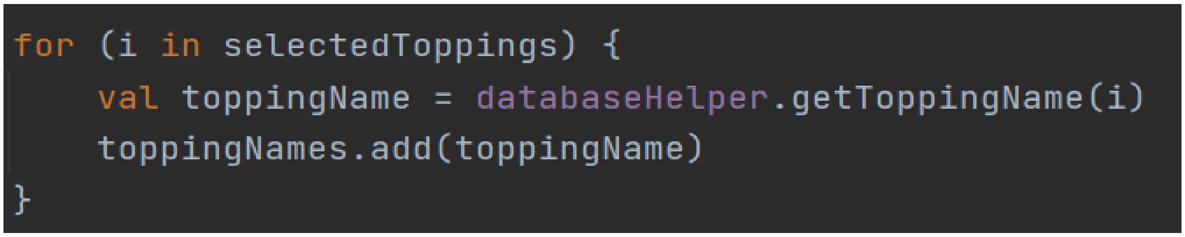
Good practice:



```
private String location;
private String description;
```

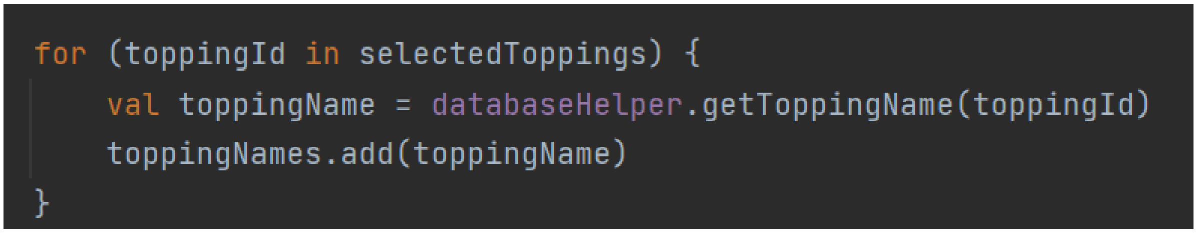
Use Intention-Revealing Names

Bad Practice:



```
for (i in selectedToppings) {
    val toppingName = databaseHelper.getToppingName(i)
    toppingNames.add(toppingName)
}
```

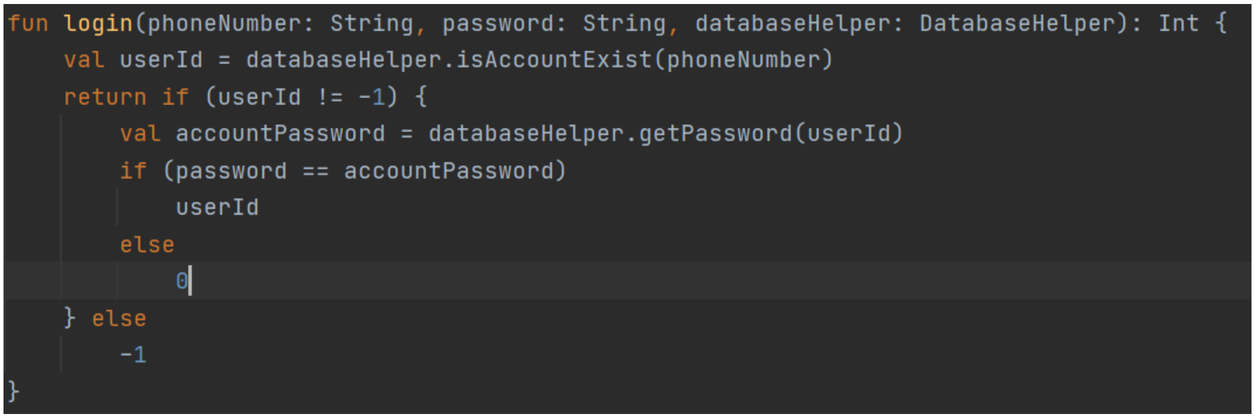
Good Practice:



```
for (toppingId in selectedToppings) {
    val toppingName = databaseHelper.getToppingName(toppingId)
    toppingNames.add(toppingName)
}
```

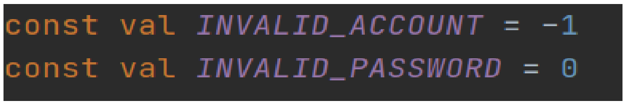
Replace Numbers with named constants:

Bad Practice:

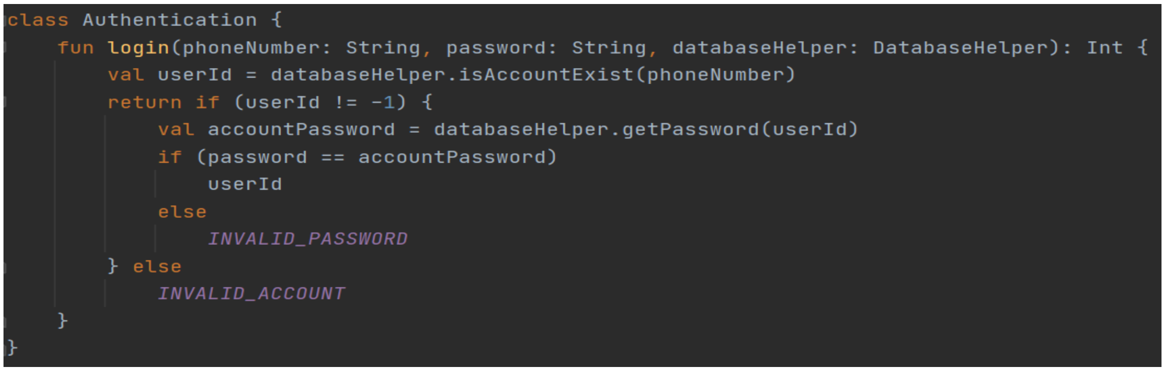


```
fun login(phoneNumber: String, password: String, databaseHelper: DatabaseHelper): Int {
    val userId = databaseHelper.isAccountExist(phoneNumber)
    return if (userId != -1) {
        val accountPassword = databaseHelper.getPassword(userId)
        if (password == accountPassword)
            userId
        else
            0
    } else
        -1
}
```

Good Practice:



```
const val INVALID_ACCOUNT = -1
const val INVALID_PASSWORD = 0
```



```
class Authentication {
    fun login(phoneNumber: String, password: String, databaseHelper: DatabaseHelper): Int {
        val userId = databaseHelper.isAccountExist(phoneNumber)
        return if (userId != -1) {
            val accountPassword = databaseHelper.getPassword(userId)
            if (password == accountPassword)
                userId
            else
                INVALID_PASSWORD
        } else
            INVALID_ACCOUNT
    }
}
```

Function naming:

A function name should start with lowercase and should be in camel case.
It should be a verb.

Bad practice:

fun GetDate()

Good Practice:

fun getDate()

Use whole words-avoid acronyms and abbreviations (unless the abbreviation is much more widely used than the long form, such as URL or HTML).

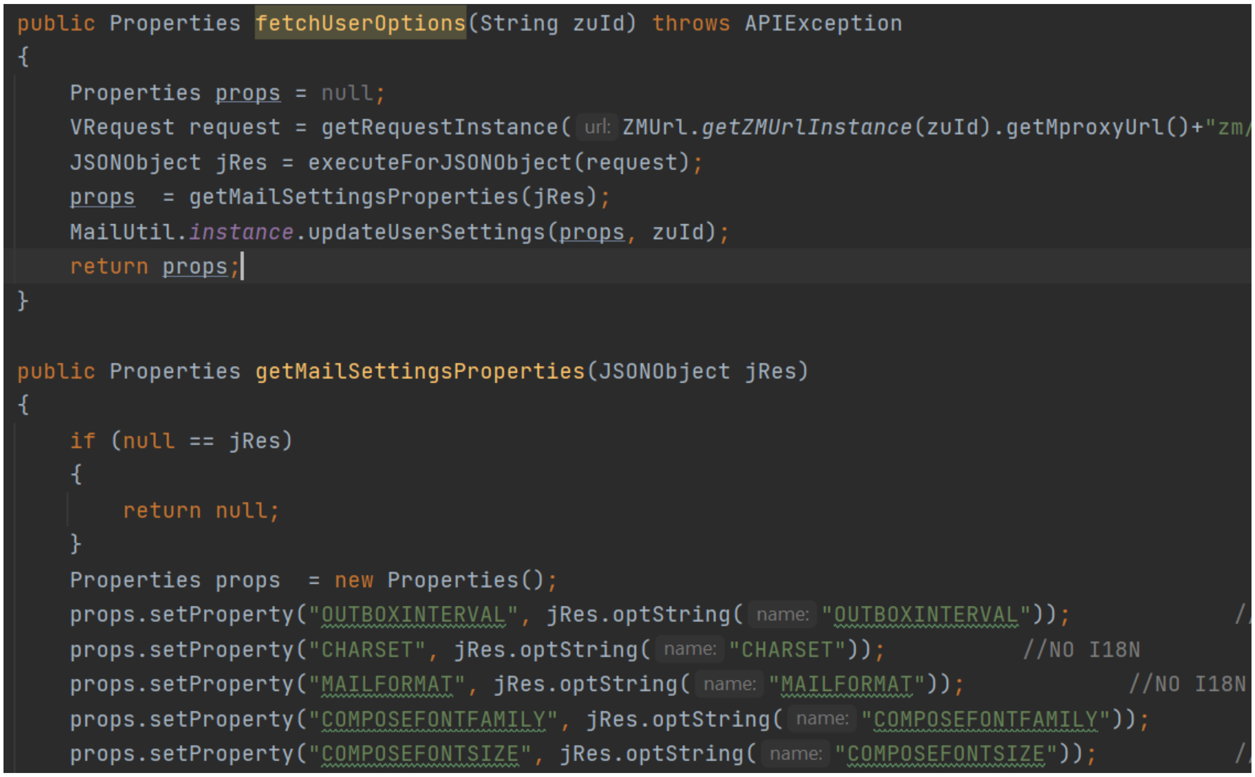
Bad Practice:

getSelectedPos()

Good Practice:
getSelectedPosition()

Pick One Word per Concept

Bad practice:



```
public Properties fetchUserOptions(String zuId) throws APIException
{
    Properties props = null;
    VRequest request = getRequestInstance( url: ZMUrl.getZMUrlInstance(zuId).getMproxyUrl()+"zm/
    JSONObject jRes = executeForJSONObject(request);
    props  = getMailSettingsProperties(jRes);
    MailUtil.instance.updateUserSettings(props, zuId);
    return props;
}

public Properties getMailSettingsProperties(JSONObject jRes)
{
    if (null == jRes)
    {
        return null;
    }
    Properties props  = new Properties();
    props.setProperty("OUTBOXINTERVAL", jRes.optString( name: "OUTBOXINTERVAL"));           /
    props.setProperty("CHARSET", jRes.optString( name: "CHARSET"));           //NO I18N
    props.setProperty("MAILFORMAT", jRes.optString( name: "MAILFORMAT"));           //NO I18N
    props.setProperty("COMPOSEFONTFAMILY", jRes.optString( name: "COMPOSEFONTFAMILY"));
    props.setProperty("COMPOSEFONTSIZE", jRes.optString( name: "COMPOSEFONTSIZE"));           /
```

Fetch and get have the same meaning so its better to use either get or fetch throughout the application.

Good practice:

```
public Properties getUserOptions(String zuId) throws APIException
```

```
public Properties getMailSettingsProperties(JSONObject jRes)
```

Class Naming:

A class should start with uppercase and should be in camel case.
It should be a noun and Do not use short forms.

Bad Practice:

Class event{ }

Good Practice:
Class Event{ }