**Intents**

**var** activityIntent = Intent(**this**, ActivityName::**class**.*java*)  
startActivity(activityIntent)

For Result

**val RATING\_ACTIVITY\_RESULT\_CODE** = 1

startActivityForResult(movieRateIntent, **RATING\_ACTIVITY\_RESULT\_CODE**)

**override fun** onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
 **super**.onActivityResult(requestCode, resultCode, data)  
  
 **if** (requestCode == **RATING\_ACTIVITY\_RESULT\_CODE**) {  
  
 **if** (resultCode == *RESULT\_OK*) {  
  
 }  
 }  
}

**In RatingActivity**  
  
**var** review = Intent()  
setResult(Activity.*RESULT\_OK*, review)  
finish()

**Menus**

**Options**

*supportActionBar*?.setDisplayHomeAsUpEnabled(**true**)-> Back button

**override fun** onCreateOptionsMenu(menu: Menu?): Boolean {  
 *menuInflater*.inflate(R.menu.*rating*, menu)  
 **return super**.onCreateOptionsMenu(menu)  
}

R.menu.rating is found in menu folder

**override fun** onOptionsItemSelected(item: MenuItem): Boolean {  
 *// Press back button*

**if** (item?.*itemId* == R.id.*MI\_submit\_rating*) {

}

**else if** (item?.itemId == android.R.id.home) {

}

**return super**.onOptionsItemSelected(item)

}

**Context**

**override fun** onCreateContextMenu(  
 menu: ContextMenu?,  
 v: View?,  
 menuInfo: ContextMenu.ContextMenuInfo?  
) {  
 **super**.onCreateContextMenu(menu, v, menuInfo)  
  
 **if** (v?.*id* == R.id.*TV\_review\_details*) {  
 *// If review has not been added* **if** (!RBar\_details\_reviews.*isIndicator*) {  
 menu?.add(1, 1001, 1, **"Add Review"**)

(groupId, itemId, orderId)  
 }  
 }  
}

**override fun** onContextItemSelected(item: MenuItem): Boolean {  
 *// If "Add Review" is pressed* **if** (item?.*groupId* == 1 && item?.*itemId* == 1001) {  
 *// Send movie details using Movie object  
 // to RatingActivity* **var** newMovie = *intent*.getSerializableExtra(**"newMovie"**) **as** Movie  
 **var** movieRateIntent = Intent(**this**, RatingActivity::**class**.*java*)  
 movieRateIntent.putExtra(**"movie"**, newMovie)  
 startActivityForResult(movieRateIntent, **RATING\_ACTIVITY\_RESULT\_CODE**)  
 }  
 **return super**.onContextItemSelected(item)  
}

**Classes**

**import** java.io.Serializable;  
  
**class** Movie (**var title**:String, **var overview**:String,**var language**:String,  
 **var releaseDate**:String, **var notSuitable**:Boolean, **var reasons**:String) : Serializable  
 {  
 **var rating**:Float = 0.0f  
 **var review**:String = **""** }

**CustListAdapter**

**internal val sList**:ArrayList<String> ?= ArrayList<String>()  
*//****TODO 3.2: Uncomment away the below statement*private val mInflater** : LayoutInflater  
  
**init** {  
 **this**.**mInflater** = LayoutInflater.from(context)  
 **sList**?.addAll(data)  
}

**override fun** getView(p0: Int, p1: View?, p2: ViewGroup?): View {  
 **val** v = **this**.**mInflater**.inflate(R.layout.*list\_item*, p2, **false**)  
 **val** label : TextView = v.findViewById(R.id.*heroname*)  
 label.*text* = **sList**?.get(p0)  
 **val** iv :ImageView = v.findViewById(R.id.*heroimage*)  
 iv.setImageResource(R.drawable.*defaulthero*)  
 **return** v  
}