**ROOM**

<https://developer.android.com/jetpack/androidx/releases/room?hl=pt-br>

dependencies {

def room\_version = "2.4.1"

implementation "androidx.room:room-runtime:$room\_version"

annotationProcessor "androidx.room:room-compiler:$room\_version"

**KAPT**

<https://kotlinlang.org/docs/kapt.html>

plugins { id('kotlin-kapt')}

dependencies { kapt("groupId:artifactId:version") }

**OU SEJA**

{ kapt("androidx.room:room-compiler:$room\_version") }

**ALTERAÇÕES NO CÓDIGO**

**MODEL PESSOA**

**@Entity**

**@Parcelize**

**data class Pessoa (**

**@PrimaryKey(autoGenerate = true) val id: Long = 0L,**

**val nome: String,**

**val peso: Double,**

**val altura: Double,**

**val imc: Double,**

**val image: String? = null**

**) : Parcelable**

**NOVO PACOTE DATABASE**

**NOVA INTERFACE PESSOADAO (PACOTE DATABASE)**

**import androidx.room.Dao**

**import androidx.room.Insert**

**import androidx.room.Query**

**import com.eamais.imc2.model.Pessoa**

**@Dao**

**interface PessoaDao {**

**@Query("SELECT \* FROM Pessoa")**

**fun buscaTodos(): List<Pessoa>**

**@Insert**

**fun salva(vararg pessoa: Pessoa)**

**}**

**NOVA CLASSE ABSTRATA APPDATABASE (PACOTE DATABASE)**

**import androidx.room.Database**

**import androidx.room.RoomDatabase**

**import com.eamais.imc2.model.Pessoa**

**@Database(entities = [Pessoa::class], version = 1)**

**@TypeConverters(Converter::class)//Caso precise fazer conversões**

**abstract class AppDatabase: RoomDatabase() {**

**abstract fun pessoaDao(): PessoaDao**

**}**

**PROBLEMAS DE CONVERSÃO?**

[**https://www.sqlite.org/datatype3.html**](https://www.sqlite.org/datatype3.html)

**NOVA CLASSE CONVERTER (PACOTE DATABASE)**

**import androidx.room.TypeConverter**

**import java.util.Date**

**class Converter {**

**@TypeConverter**

**fun paraTimestamp(value: Long?): Date? {**

**return value?.*let* { Date(it) }**

**}**

**@TypeConverter**

**fun dateParaTimestamp(date: Date?): Long? {**

**return date?.*time*?.toLong()**

**}**

**}**

**ALTERAÇÃO NO FORMULÁRIO (ACTIVITY)**

**Log.i("Form\_Pessoa\_Activity", "OnCreate: $pessoa")**

**val db = Room.databaseBuilder(**

**this,**

**AppDatabase::class.*java*,**

**"imc.db"**

**).allowMainThreadQueries().build()**

**val pessoaDao = db.pessoaDao()**

**pessoaDao.salva(pessoa)**

**finish()**

**ALTERAÇÕES APPDATABASE**

**@Database(entities = [Pessoa::class], version = 1)**

**abstract class AppDatabase: RoomDatabase() {**

**abstract fun pessoaDao(): PessoaDao**

**companion object {**

**fun instancia(context: Context) : AppDatabase{**

**return Room.databaseBuilder(**

**context,**

**AppDatabase::class.*java*,**

**"imc.db"**

**).allowMainThreadQueries().build()**

**}**

**}**

**}**

**ALTERAÇÕES MAIN**

**override fun onResume() {**

**super.onResume()**

**val db = AppDatabase.instancia(this)**

**val pessoaDao = db.pessoaDao()**

**adapter.refreshAll(pessoaDao.buscaTodos())**

**}**

**DELETAR PACOTE DAO E REFATORAR CÓDIGO!**

**class ListaAdapter (**

**pessoas: List<Pessoa> = *emptyList*(),**

**private val context: Context,**

**var click: (pessoa: Pessoa) -> Unit = {})**

**class MainActivity : AppCompatActivity() {**

**private val adapter = ListaAdapter(context = this)**

**MENUS**

[**https://developer.android.com/guide/topics/ui/menus?hl=pt-br**](https://developer.android.com/guide/topics/ui/menus?hl=pt-br)

**NOVO ANDROID RESOURCE DIRECTORY -> MENU**

**NOVO MENU -> MENU\_DETAIL**

**<?xml version="1.0" encoding="utf-8"?>**

**<menu xmlns:app="http://schemas.android.com/apk/res-auto"**

**xmlns:android="http://schemas.android.com/apk/res/android">**

**<item**

**android:id="@+id/menu\_editar"**

**android:icon="@drawable/ic\_action\_edit"**

**android:title="Editar"**

**app:showAsAction="ifRoom" />**

**<item**

**android:id="@+id/menu\_delete"**

**android:icon="@drawable/ic\_action\_delete"**

**android:title="Deletar"**

**app:showAsAction="ifRoom" />**

**</menu>**

**ACESSANDO DADOS COM DAO**

[**https://developer.android.com/training/data-storage/room/accessing-data?hl=pt-br#:~:text=Mantenha%20tudo%20organizado%20com%20as,com%20base%20nas%20suas%20prefer%C3%AAncias.&text=A%20fim%20de%20interagir%20com,DAOs%20na%20sigla%20em%20ingl%C3%AAs**](https://developer.android.com/training/data-storage/room/accessing-data?hl=pt-br#:~:text=Mantenha%20tudo%20organizado%20com%20as,com%20base%20nas%20suas%20prefer%C3%AAncias.&text=A%20fim%20de%20interagir%20com,DAOs%20na%20sigla%20em%20ingl%C3%AAs)**).**

**ALTERAÇÃO MAIN**

**private fun configRecyclerView() {**

**val rview = binding.rview**

**rview.*adapter* = adapter**

**adapter.click = { pessoa ->**

**val intent = Intent(**

**this, DetailPessoaActivity::class.*java***

**).*apply* {**

**putExtra("PESSOA\_ID", pessoa.id)**

**}**

**startActivity(intent)**

**}**

**}**

**ALTERAÇÕES**

**class DetailPessoaActivity : AppCompatActivity() {**

**private var pessoaId: Long = 0L**

**private var pessoa: Pessoa? = null**

**private val binding by *lazy* {**

**ActivityDetailPessoaBinding.inflate(*layoutInflater*)**

**}**

**private val pessoaDao by *lazy* {**

**AppDatabase.instancia(this).pessoaDao()**

**}**

**override fun onCreateOptionsMenu(menu: Menu?): Boolean {**

***menuInflater*.inflate(R.menu.*menu\_detail*, menu)**

**return super.onCreateOptionsMenu(menu)**

**}**

**override fun onOptionsItemSelected(item: MenuItem): Boolean {**

**val db = AppDatabase.instancia(this)**

**val pessoaDao = db.pessoaDao()**

**when (item.*itemId*) {**

**R.id.*menu\_delete* -> {**

**Log.i("DetailPessoaActivity", "OnOptionsItemSelected: remover")**

**pessoa?.*let* { pessoaDao.deleta(it) }**

**finish()**

**}**

**R.id.*menu\_editar* -> {**

**Log.i("DetailPessoaActivity", "OnOptionsItemSelected: editar")**

**Intent(this, FormPessoaActivity::class.*java*).*apply* {**

**putExtra("PESSOA\_ID", pessoaId)**

**startActivity(this)**

**}**

**}**

**}**

**return super.onOptionsItemSelected(item)**

**}**

**override fun onCreate(savedInstanceState: Bundle?) {**

**super.onCreate(savedInstanceState)**

**setContentView(binding.*root*)**

**pessoaId = *intent*.getLongExtra("PESSOA\_ID", 0L)**

**}**

**override fun onResume() {**

**super.onResume()**

**pessoa = pessoaDao.buscaId(pessoaId)**

**pessoa?.*let* {**

**preencheCampos(it)**

**} ?: finish()**

**}**

**private fun preencheCampos(pessoa: Pessoa) {**

**...**

**}**

**}**

**}**

**class FormPessoaActivity : AppCompatActivity() {**

**private val binding by *lazy*{**

**ActivityFormPessoaBinding.inflate(*layoutInflater*)**

**}**

**private var url: String?= null**

**private var idPessoa = 0L**

**override fun onCreate(savedInstanceState: Bundle?) {**

**super.onCreate(savedInstanceState)**

**setContentView(binding.*root*)**

**val botao = binding.botao**

**botao.setOnClickListener {**

**...**

**val pessoa = Pessoa(**

**id = idPessoa,**

**nome = nome,**

**peso = peso,**

**altura = altura,**

**imc = peso/(altura \* altura),**

**image = url**

**)**

**Log.i("Form\_Pessoa\_Activity", "OnCreate: $pessoa")**

**val db = AppDatabase.instancia(this)**

**val pessoaDao = db.pessoaDao()**

**if (idPessoa > 0){**

**pessoaDao.atualiza(pessoa)**

**} else{**

**pessoaDao.salva(pessoa)**

**}**

**pessoaDao.salva(pessoa)**

**finish()**

**}**

***intent*.getParcelableExtra<Pessoa>(*CHAVE\_PESSOA*)?.*let* {**

***title* = "Editar Pessoa"**

**idPessoa = it.id**

**url = it.image**

**binding.editImage.*loadImag*(it.image)**

**binding.editNome.setText(it.nome)**

**binding.editPeso.setText(it.peso.toString())**

**binding.editAltura.setText(it.altura.toString())**

**}**

**...**

**}**

**}**

**@Dao**

**interface PessoaDao {**

**@Query("SELECT \* FROM Pessoa")**

**fun buscaTodos(): List<Pessoa>**

**@Insert**

**fun salva(vararg pessoa: Pessoa)**

**@Update**

**fun atualiza(vararg pessoa: Pessoa)**

**@Delete**

**fun deleta(vararg pessoa: Pessoa)**

**}**