

«Extends AppCompatActivity» LoginActivity
- signInButton: SignInButton - mGoogleSignInClient: GoogleSignInClient - TAG: String - mAuth: FirebaseAuth - btnSignIn: Button - RC_SIGN_IN: int
+ signIn(): # onCreate(savedInstanceState: Bundle): # onActivityResult(requestCode: int, data: Intent): - handleSignInResult(completedTask: Task<GoogleSignInAccount>): - FirebaseAuth(acct: GoogleSignInAccount): - updateUI(fUser: FirebaseUser):

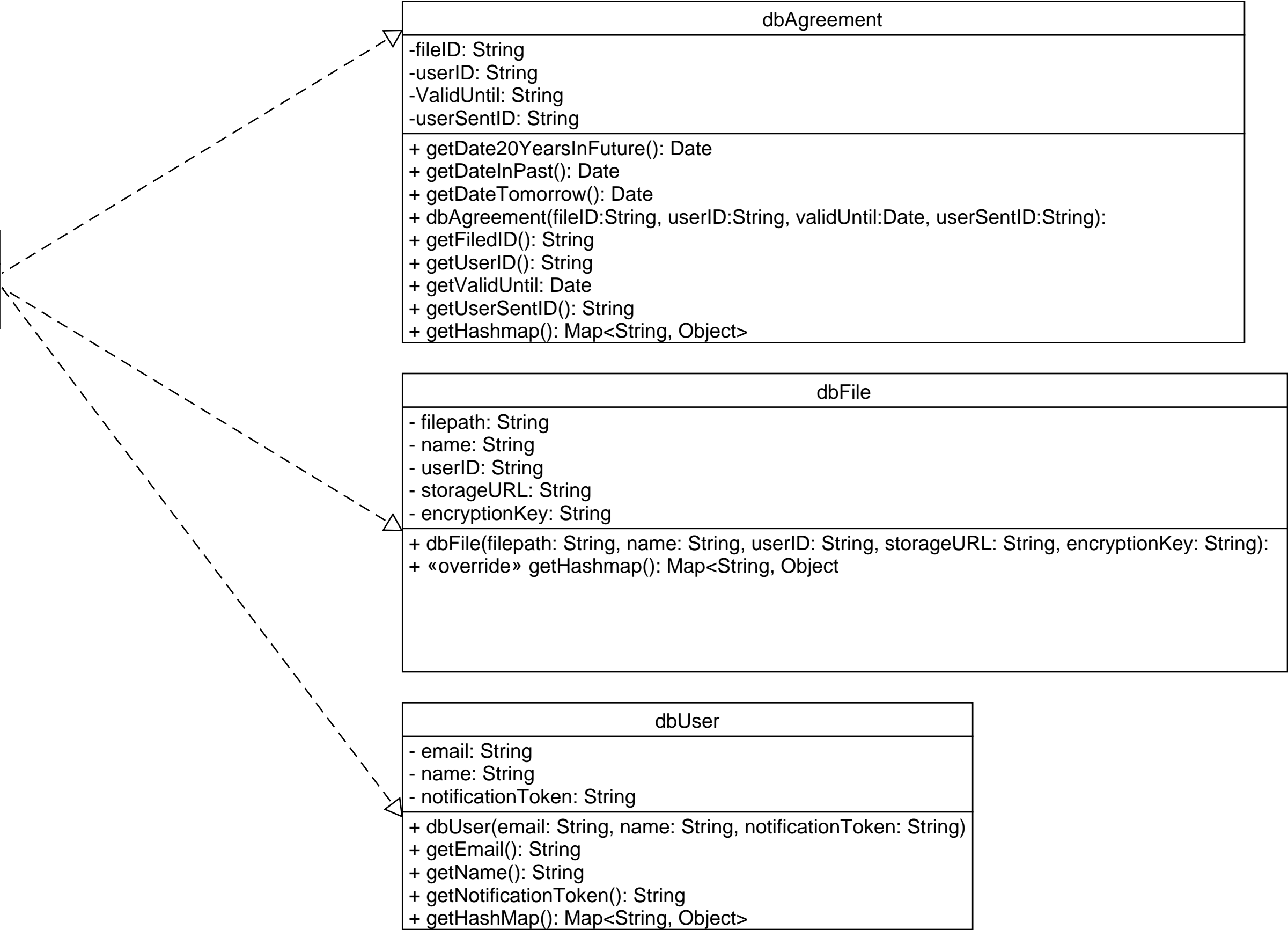
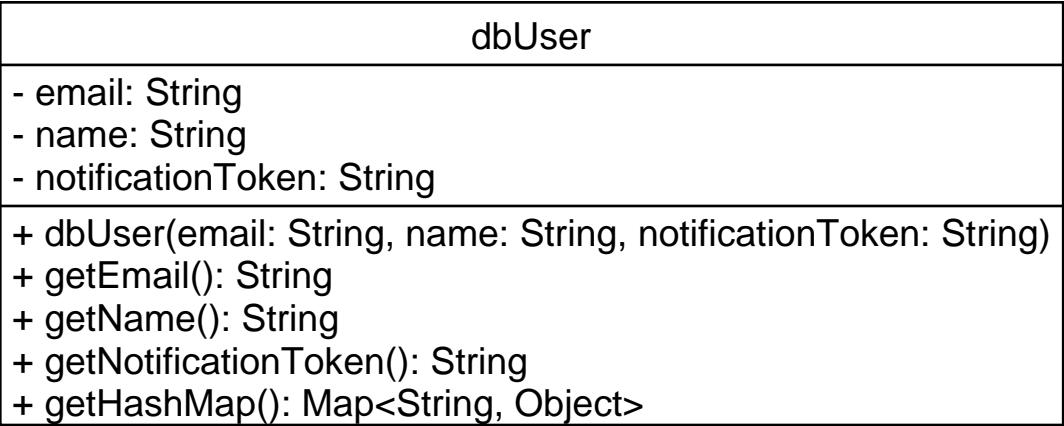
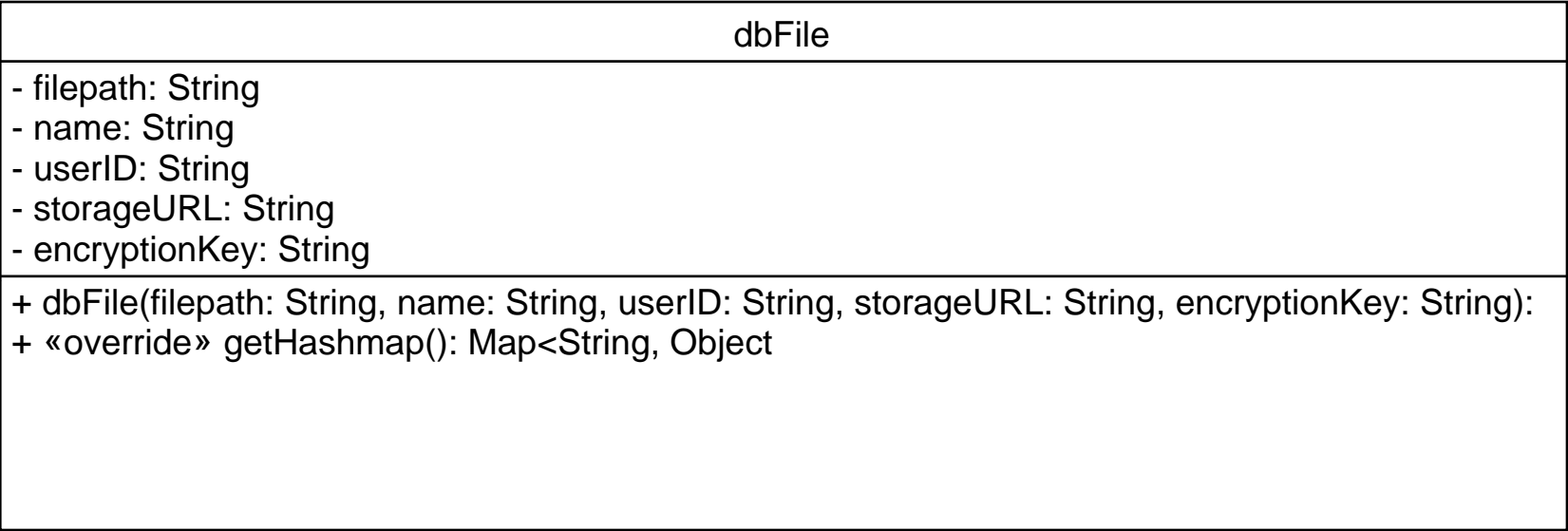
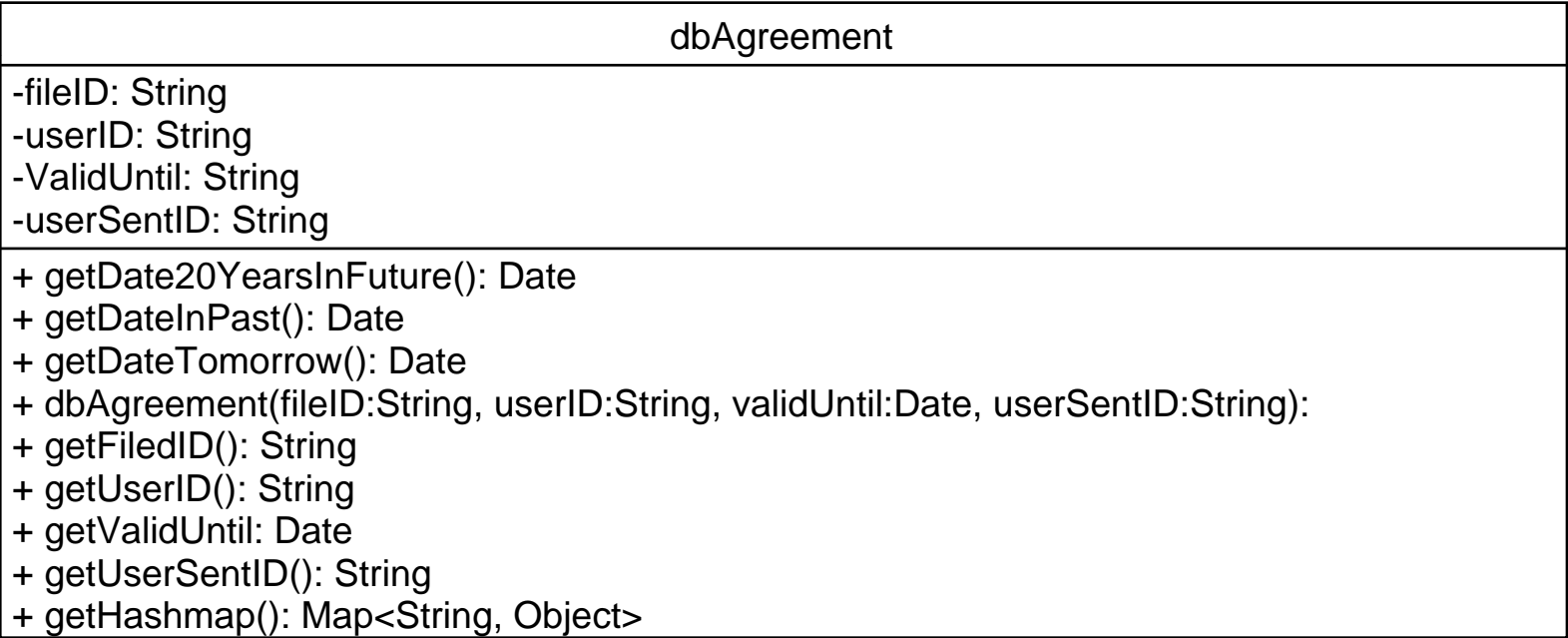
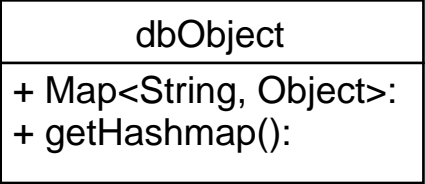
«Extends BaseActivity» SendFileActivity
- «final» FILE_RESULT_SUCCESS: int - «final» FILE_REQUEST_CODE: int - «final» LOG_TAG: String - storage: FirebaseStorage - auth: FirebaseAuth - btnChooseFile: Button - btnChooseRecievedFile: Button - btnSend: Button - txtEmail: EditText - txtFilename: EditText - lblFilename: TextView - progressBar: ProgressBar - errorHandler: MyError - file: MyFile - isSendFileLocalStorage: boolean - fileToSend: dbFile - userThatSentFile: dbUser - determineCurrentUser()
«Implements View.OnClickListener» chooseReceivedFile
+ «override» onClick(view: View): - makeDialogBox():
«Implements View.OnClickListener» «private» sendFile
- filePathFirebase: String - filename: String - user: FirebaseUser - userToReceiveID: String - fileRef: StorageReference - + «override» onSuccess(data: Map<String, Object>, message: String): + «override» onFailure(error: String, errorCode: MyError.ErrorCode): - afterGetEmail(data: Map<String, Object>, message: String): - afterUploadFile(taskSnapShot: UploadTask.TaskSnapshot, key: String): - afterUpdateDB(): - afterFailGetEmail(error: String, errorCode: MyError.ErrorCode): - afterFailUpload(exception: Exception): - afterFailUpdateDB(e: Exception): - cleanUp():
«Implements TextWatcher» FileNameChange
+ «override» beforeTextChanged(charSequence: CharSequence, i: int, i1: int, i2: int): + «override» onTextChanged(charSequence: CharSequence, i: int, i1: int, i2: int):

«Extends AppCompatActivity» MainActivity
+ onCreate(savedInstanceState: Bundle, persistentState: PersistableBundle):

SayHello
+ hello(name: String): String

«Extends BaseActivity» MySentFiles
«final» USER_COLLECTION_NAME: String «final» FILE_COLLECTION_NAME: String - LOG_TAG: String - currentFileSelected: SingleSentFile - errorHandler: MyError + btnApproved: Button + btnAll: Button + btnPending:Button + adapter: MyRecyclerViewAdapter - details: HashMap<String, HashMap<String, Object>>()
getAsync(collectionName: String, docID: String, cb: Callback): + «override» onCreate(savedInstanceState: Bundle): + recyclerViewStuff(): + showPopup(v: View): + colourButtons(clicked: Button): + doButtons(): + changePermissions(d: Date, file: SingleSentFile): + revoke(file: SingleSentFile): + approveTilTomorrow(file: SingleSentFile): + approveIndefinitely(file: SingleSentFile:
«Extends RecyclerView.Adapter<Holder>» MyRecyclerViewAdapter
+ MyRecyclerViewAdapter(data: QuerySnapshot) -«final» LOG_TAG: String - mDataSet: ArrayList<SingleSentFile> - filtered: ArrayList<SingleSentFile> - currentFilter: Filter<SingleSentFile> + «override» onCreateViewHolder(group: ViewGroup, i: int): Holder + «override» onBindViewHolder(holder: Holderm position: int): + «overide» getItemCount(): int + filter(filter: Filter<SingleSentFile>): + filterWithCurrentFilter():

«Extends BaseActivity» RecieveFilesActivity
+ «final» USER_COLLECTION_NAME: String + «final» FILE_COLLECTION_NAME: String + btnSort: Button + editFilter: EditText + db: FirebaseFirestore + myAdapter: RecyclerAdapter + user: FirebaseUser + mFirebaseFunctions: FirebaseFunctions + «final» LOG_TAG: String + «override» onCreate(savedInstanceState Bundle): + setEvents(): + setElements(): - determineCurrentUser(): # getAsync(collectionName: String, docID: String, cb: Callback): - setUpFireStore(): - setUpRV(): - sendNotificationRequestingPermission(fName: String, ownerToken: String): - showHamburgerPopUp(v: View, position: int, file: FileModel) # getAsync(collectionName: String, docID: String, cb: Callback): - setUpFireStore(): - setUpRV(): - showHamburgerPopUp(v: View, position: int, file: FileModel)
«Implements View.OnClickListener» Sorter
+ «override» onClick(v: View): + showSortPopup(v: View):
«Implements TextWatcher» Filter
+ «override» beforeTextChanged(charSequence: CharSequence, i: int, i1: int, i2: int): + «override» onTextChanged(charSequence: CharSequence, i: int, i1: int, i2: int): + «override» afterTextChanged(editable: Editable):
«Extends RecyclerView.Adapter<RecyclerViewHolder> » RecyclerViewAdapter
+ «override» onCreateViewHolder(group: ViewGroup, i: int): Holder + «override» onBindViewHolder(holder: Holderm position: int):
«Extends RecycklerView.Adapter<RecyclerViewHolder>» RecyclerViewAdapter
(Variables)
(Functions)



AES encryption

- secretKey: SecretKey

+ AES encryption(length: int)
+ AES encryption()
+ AES encryption(key: String):
+ encrypt(stream: InputStream): InputStream
+ decrypt(stream: InputStream): InputStream
+ getKey(): String

RSA

- privateKey: PrivateKey
- publicKey: PublicKey

+ RSA(keylength: int):
+ RSA():
+ RSA(privateKey: String, publicKey: String):
+ encrypt(stream: InputStream): InputStream
+ decrypt(encrypted: InputStream): InputStream
+ getPrivateKey(): String
+ getPublicKey(): String