oo::Person ~plz, email, vorname, nachname, passwort, datum: String ~personenID: int sql\_generator::Main ~fileName = "SQL.txt": String +Person(String plz, String email, String vorname, String nachname, String passwort, String datum, int personenID): ctor +getRandomIntegerBetweenRange(int min, int max): int +Person(String plz, String email, String vorname, String nachname, String passwort, String datum, int personenID, String strasse, int hausnummer): ctor +main(String[] args): void +generiereSitze(int saal, int reihen, int spalten): void +getStrasse(): String +setStrasse(String strasse): void exception::RequiredFactoryFailedException exception::ResultSetIsNullException exception.registrierung::UserAlreadyExistsException exception::FailedObjectCreationException exception::InvalidInputValueException +getHausnummer(): int serAlreadvExistsException(): ctor +FailedObjectCreationException(): ctor +RequiredFactoryFailedException(): ctor -ResultSetIsNullException(): ctor /alidInputValueException(): ctor +setHausnummer(int hausnummer): void +getPlz(): String +setPlz(String plz): void +getEmail(): String +setEmail(String email): void +getVorname(): String helper::DateFormatter helper::ArrayBuilder -frontendDate = new SimpleDateFormat("EEE, dd. MMM", new Locale("de", "DE")): SimpleDateFormat helper::ExceptionHandler -frontendTime = new SimpleDateFormat("HH:mm", new Locale("de", "DE")): SimpleDateFormat +stringToIntArray(String string, String del): int[] +stringToIntArray(String string): int[] -sqlDate = new SimpleDateFormat("yyyy-MM-dd"): SimpleDateFormat exception::EmptyResultSetException -sqlTime = new SimpleDateFormat("HH:mm:ss"): SimpleDateFormat exception::FailedDataInsertionException exception.registrierung::EmptyInputValueException exception.registrierung::UnmatchingPasswordException exception::UnequalParameterLengthException +setVorname(String vorname): void +intArrayToString(int[] arr, String del): String
+intArrayToString(int[] arr): String +getNachname(): String
+setNachname(String nachname): void -sqlDateAndTime = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss"): SimpleDateFormat nmatchingPasswordException(): ctor +FailedDataInsertionException(): ctor +UnequalParameterLengthException(): ct EmptyInputValueException(): cto +getFrontendDate(Date date): String helper::PLZFormatter +getPasswort(): String +setPasswort(String passwort): void +getFrontendTime(Date date): String +addLeadingZeros(int plz): Stri +getSQLDate(Date date): String +getDatum(): String +setDatum(String datum): void +getSQLTime(Date date): String +getSQLDateAndTime(Date date): String helper::SupportMethods +getPersonenID(): int helper::SeatIDFormatter +getResultSetSize(ResultSet rs): int +setPersonenID(int personenID): void +removeHTMLCode(String input): String +seatsStringToIntArray(String seatIDString): int[] +close(Connection c, ResultSet rs): void +close(Connection c): void +close(ResultSet rs): void +removeSQLInjections(String input): String helper::TempBuchungHandler factory::ReservierungsFactory +addTempBuchungToSession(HttpSession session, String sitzIDs, String preisVerIDs, int vorstellungsID): void +createReservierungsBelege(int[] sitzeIDs, int[] preiseVerIDs, String seats, String preisVer, int vorstellungsID, int KNR): int +getReservierungsbelegByRNR(int RNR): Reservierungsbeleg +getReservierungsbelegByKID(int KID): Reservierungsbeleg[] +createReservierungsPositionen(Connection c, int RNR, Sitz[] sitze, int[] preiseVerIDs): void +createReservierungsBelegPDF(int KID, Vorstellung vorstellung, Sitz[] sitze): void oo::Kunde oo::Admin ~kundenID, treuepunkte: int +Admin(String plz, String email, String vorname, String nachname, String passwort, String datum, int personenID, int mitarbeiterID): ctor factory::SitzsperreFactory +Kunde(String plz, String email, String vorname, String nachname, String passwort, String datum, int personenID, String strasse, int hausnummer, int kundenID, int treuepunkte): ctor -minutesUntilTimeOut = 10: int +Admin(): ctor +lockSeats(int[] seatIDs, int vorstellungsID, int KID): Sitzsperre[] +setKundenID(int kundenID): void +getLockedSeats(int vorstellungsID): Sitzsperre[] +setMitarbeiterID(int mitarbeiterID): vo +getTreuepunkte(): int +updateLockedSeats(): Sitzsperre[] +setTreuepunkte(int treuepunkte): void +getTimedOutSeats(): Sitzsperre[]
+deleteTimedOutSeats(Sitzsperre[] s): Sitzsperre[] +locSeats(Sitz[] seats, Vorstellung v, int KID): Sitzsperre[] +getLockedSeats(Vorstellung v): Sitzsperre[] oo::Régistrierung Password::PassMD5 db\_connector::Connector factory::VorstellungsFactory +hash(String pass): String ~vorname: String -sdate = "": String ~geburtsdatum: Date +executeQuery(Connection c, String sql): void -stime = "": String ~email: String +getQueryResult(Connection c, String sql): ResultSet ~passwordHash: String +closeConnection(Connection c): void +closeResultSet(ResultSet rs): void +getVorstellungen(Film film, String date, String time, String plz, ResultSet mockRs): Vorstellung[] +getVorstellungen(Film film, String date, String time, String plz): Vorstellung[] +getVorstellungByld(int id, ResultSet mockRs): Vorstellung öö::UserLogin ~adresszusatz: Stri +getVorstellungByld(int id): Vorstellung +getLastSQLQuery(): String +Registrierung(String vorname, String nachname, Date geburtsdatum, String email, String passwordHash, int plz, String straße, int hausnummer, String adresszusatz): ctor -firstname: Strir +Registrierung(int PID, String vorname, String nachname, Date geburtsdatum, String email, String passwordHash, int plz, String straße, int hausnummer, String adresszusatz): ctor -lastname: String +getVorname(): String +setVorname(String vorname): void +getNachname(): String factory::FilmFactory +UserLogin(String email, String firstname, String lastname, int PID, int KID): ctor +setNachname(String nachname): voi +getFirstname(): String +getGeburtsdatum(): Date +getFilme(String search, String date, String time, int fsk, String plz, ResultSet mockRs1, ResultSet mockRs2, ResultSet mockRs3): Film[] +setFirstname(String firstname): void +setGeburtsdatum(Date geburtsdatum): vo +getFilme(String search, String date, String time, int fsk, String plz): Film[] +getEmail(): String +getLastname(): String +getFilme(String search, String date, String time, int fsk, String plz, int genreID, ResultSet mockRs1, ResultSet mockRs2, ResultSet mockRs3): Film[] oo::ReservierungsStornierung +setEmail(String email): void +setLastname(String lastname): void +getPasswordHash(): String +getFilme(String search, String date, String time, int fsk, String plz, int genreID): Film[] +setPID(int PID): void +setPasswordHash(String passwordHash): void 🗸 +getTitelPageFilme(String plz, ResultSet mockRs1, ResultSet mockRs2, ResultSet mockRs3): Film[] +ReservierungsStornierung(int RNR, int strnNr): ctor +getTitelPageFilme(String plz): Film[]
+getFilm(int id, ResultSet mockRs1, ResultSet mockRs2, ResultSet mockRs3): Film +setPlz(int plz): void +setKID(int KID): void +getStraße(): String +setRNR(int RNR): void +getEmail(): String +setEmail(String email): void +getStrnNr(): int -setSprachen(Film film, ResultSet mockRs): void +getHausnummer(): int +setStrnNr(int strnNr): void -setGenres(Film film, ResultSet mockRs): void +setHausnummer(int hausnummer): void +getAdresszusatz(): String +setAdresszusatz(String adresszusatz): voic factory::SitzFactory +setPID(int PID): void db\_connector::QueryBuilder +getSitzById(int id, ResultSet mockRs): Sitz +getKundeByPID(int PID): String +getKundeByKID(int KID): String -getBookedSeats(int vorstellungsID): Sitz[] +getAllLockedSeats(int vorstellungsID): Sitz oo::ReservierungsPosition +createLoginQuery(String email, String passwordHash): String -sitzplatzID: in -KNR: int -getReservedSeats(int vorstellungsID): Sitz +createUser(String name, String lastname, Date gebDate, String email, String passwordHash, int hausnummer, String straße, String adresszusatz, int plz): String +createUser(String name, String lastname, Date gebDate, String email, String passwordHash, int hat high was the showAllCinemas(): String high was the showCinemalmaginationsToday(): String high was the showCinemalmaginationsThisWeek(): String high was the showTitlePageFilms(): String high was the showAllFilmInfos(String filmTitel): String high was the showGenresForFilmTitle(String filmTitel): String high was the showGenresForFilmID(int filmID): String high was the showGenresForFilmID(int filmID): String high was the showGenresForFilmID(int filmID): String high was the showGenresCouncery(String search, String date, String time, int fsk, String plz): String high high was the showMovieById(String search, String date, String time, int fsk, String plz, int genreID): String high high was the showMovieById(String id, String date, String time, String plz): String high high was the show many string high +getSitzeByBNR(int BNR): Sitz[] -vorstellungsID: in +getSitzeByRNR(int RNR): Sitz[] oo::Ort -timestamp: Date +getSitzById(int id): Sitz oo::Reservierungsbeleg → +ReservierungsPosition(int positionsID, int RNR, int sitzID): ctor +Sitzsperre(int sitzplatzID, int vorstellungsID, int KNR, Date timestamp): cto +getPositionsID(): int թթաւթյանը (int belegID, float preis, Vorstellung vorstellung, Kunde kunde, Date uhrzeit)։ ctor +getSitzplatzID(): int +Ort(String ortsName, int plz): ctor +setPositionsID(int positionsID): void +setSitzplatzID(int sitzplatzID): void +getBNR(): int +getOrtsName(): String +getKNR(): int factory::BuchungsStornierungFactory +setBNR(int BNR): void +setOrtsName(String ortsName): void +setKNR(int KNR): void +createStornierung(int BNR): BuchungsStornierung +getSitzID(): int +getTimestamp(): Date +getBuchungsStornierungByStrnNR(int StrnNR): BuchungsStornierung
+getBuchungsStornierungByBNR(int BNR): BuchungsStornierung +setSitzID(int sitzID): void +setPlz(int plz): void +setTimestamp(Date timestamp): void +getBuchungsStornierungByKID(int KID): BuchungsStornierung[] ~preis: float factory::GebaeudeFactory ~vorstellung: Vorstellung +getGebaeude(ResultSet mockRs): Gebaeude[] ~kunde: Kunde +getGebaeude(): Gebaeude[] oo::Gebaeude +getGebaeudeById(int id): Gebaeude oo::Kinosaal +Beleg(int belegID, float preis, Vorstellung vorstellung, Kunde kunde, Date uhrzeit): ctor +getGrundPreis(int id): String +getPreisveränderungen(): String ~strasse: String -sitzplatzID, nummer: int ~bezeichnung: String ~hausnummer: int -reihe, sitzklasse: char +getPreisverAnderungen(): String
+getPreiseLaenge(): String
+getVorstellungByIdPLZ(int id): String
+getSitzById(int id): String
+createBuchungsBeleg(int KID, int vorstellungsID, float preis, String timestamp): String
+createBuchungsposition(int posID, int BNR, int sitzID): String
+createPreisĤnderungBuchung(int posID, int BNR, int preisVerID): String
+aendereProfil(String vorname, String nachname, String email, String passwort, String strasse, String hausnummer, int PID): String
+getBuchungsbelegByKIDandTimestamp(int KID, String timestamp): String
+getTimedOutSitzsperre(int minuteTimeDiff, int id): String
+deleteTimedOutSitzSperre(int minuteTimeDiff): String
+createSitzsperre(int sitzplatzID, int vorstellungsID, int KID, Date date): String +getVorstellung(): Vorstellung +getKunde(): Kunde +getUhrzeit(): Date ~gebaeude: Gebaeude +Sitz(int sitzplatzID, int nummer, char reihe, char sitzklasse): ctor ~ortsname: String factory::AnfahrtsseiteFactory +getSitzplatzID(): int +Kinosaal(int saalID, String bezeichnung, Sitz[] sitzplan, Gebaeude gebaeude): ctor +Gebaeude(int gebäudeld, String strasse, int hausnummer, int plz, String ortsname): ctor +setBelegID(int belegID): void +setSitzplatzID(int sitzplatzID): void +getAnfahrtsseite(String ort, ResultSet mockRs): String +getGebaeude(): Gebaeude +getOrtsname(): String +setPreis(float preis): void +getNummer(): int +setGebaeude(Gebaeude gebaeude): void +getAnfahrtseite(String ort): String +getGebäudeld(): int +setNummer(int nummer): void +setVorstellung(Vorstellung vorstellung): void +getStrasse(): String ı +setKunde(Kunde kunde): void +getReihe(): char +getBezeichnung(): String +getSitzplan(): Sitz[] +setSaalID(int saalID): void +getHausnummer(): int +setReihe(char reihe): void +setUhrzeit(Date uhrzeit): void factory::KinosaalFactory +getSitzklasse(): char +setGebäudeld(int gebäudeld): void +setSitzklasse(char sitzklasse): voic +setBezeichnung(String bezeichnung): void +setSitzplan(Sitz[] sitzplan): void +getKinosaal(int id, ResultSet mockRs1, ResultSet mockRs2): Kinosaal +setStrasse(String strasse): void +delete TimedOutSitzSperre(int minute TimeDiff): String
+createSitzsperre(int sitzplatzID, int vorstellungsID, int KID, Date date): String
+getBuchungsbelegByBNR(int BNR): String
+getBuchungsbelegeByKID(int KID): String
+createReservierungsbelegGint KID, int vorstellungsID, float preis, String timestamp): String
+getReservierungsbelegByKIDandTimestamp(int KID, String timestamp): String
+createReservierungsposition(int posID, int RNR, int sitzID): String
+createPreisĤnderungReservierung(int posID, int RNR, int preisVerID): String
+getReservierungsbelegByKID(int KID): String
+getReservierungsbelegByRNR(int RNR): String
+getUserForRegistration(String firstname, String lastname, String gebDate, String email): String
+getUserByEmail(String email): String
+getUserByEmail(String email): String
+getGebaeudeById(int id): String
+getBookedSeats(int vorstellungsID): String
+getBookedSeats(int vorstellungsID): String
+getReservedSeats(int vorstellungsID): String
+getReservierungsPositionenByRNR(int RNR): String
+getReservierungsPositionenByRNR(int RNR): String
+getJustCreatedBuchung(int KID): String
+getJustCreatedReservierung(int KID): String
+getJustCreatedReservierung(int KID): String
+createBuchungsStornierung(int BNR): String √ +getKinosaal(int id): Kinosaal +setHausnummer(int hausnummer): void +getRowLength(char row): int +setPlz(int plz): void +setOrtsname(String ortsname): void factory::BuchungspositionFactory +getBuchungspositionenByBNR(int BNR): BuchungsPosition[] oo::BuchungsPosition oo::Vorstellung -BNR: int factory::RegistrierungFactory ~vorstellungsID: in +createRegistrierung(String vorname, String nachname, Date geburts atum, String email, String passwordHashWdh, String wohnort, int plz, String straße, int hausnummer, String adresszusatz): Registrierung -sprache: String ~film: Film oo::Buchungsbeleg +BuchungsPosition(int positionsID, int BNR, int sitzID): ctor +getRegistrierungByEmail(String email): Registrierung +getPositionsID(): int +Buchungsbeleg(int belegID, float preis, Vorstellung vorstellung, Kunde kunde, Date uhrzeit): ctor ~saal: Kinosaal +setPositionsID(int positionsID): void ~datum: Date factory::KundenFactory +Vorstellung(int vorstellungsID, Date datum, Date uhrzeit, String sprache, Film film, Kinosaal saal): ctor +getSitzID(): int / +setSitzID(int sitzID): void +getVorstellungsID(): int +getKunde(int PID, ResultSet mockRs): Kunde +setVorstellungsID(int vorstellungsID): void +getDatum(): Date +qetKundeByKID(int KID, ResultSet mockRs): Kunde +setDatum(Date datum): void +getJustCreatedReservierung(int KID): String
+createBuchungsStornierung(int BNR): String
+createReservierungsStornierung(int RNR): String
+getBuchungsStornierungByStrnNR(int StrnNR): String
+getReservierungsStornierungByRNR(int RNR): String
+getReservierungsStornierungByStrnNR(int StrnNR): String
+getBuchungsStornierungByKID(int KID): String
+getReservierungsStornierungByKID(int KID): String
-getDateAsString(): String
-getTimeAsString(): String +getKundeByKID(int KID): Kunde +getUhrzeit(): Date +setUhrzeit(Date uhrzeit): void +getSprache(): String +setSprache(String sprache): void +getFilm(): Film factory::ReservierungspositionFactory +getReservierungsPositionenByRNR(int RNR): ReservierungsPosition[ +getSaal(): Kinosaal +setSaal(Kinosaal saal): void qr\_code::QrCodeGenerator oo::BuchungsStornierung QR\_CODE\_IMAGE\_PATH = "../../GitProjekte/CineflexV1/web/img/qrcode/MyQRCode.png;": String factory::PreisänderungsFactory +generateQRCodeImage(String text, String filepath): void +getPreisänderungByld(int preisänderungsID): Preisänderung -strnNr: int +main(String[] args): void +BuchungsStornierung(int BNR, int strnNr): ctor  $\triangle$  +getBNR(): int ~titel, beschreibung, bildLink, trailerLink: Strin pdf\_generator::PdfGenerator +getStrnNr(): int +setStrnNr(int strnNr): void factory::GenreFactory ~dauer, fsk, filmID: int +createBuchungsPDF(String pathPDF, String pathQR, Buchungsbeleg buchungsbeleg, Vorstellung vorstellung, Sitz[] sitze, Kunde kunde): void +getGenreByld(int id, ResultSet mockRs): Genre +createReservierungsPDF(String pathPDF, String pathQR, Reservierungsbeleg reservierungsbeleg, Vorstellung vorstellung, Sitz[] sitze, Kunde kunde): void +getGenreById(int id): Genre +getGenre(ResultSet mockRs): Genre[] +Film(String titel, String beschreibung, String bildLink, String trailerLink, int dauer, int fsk, int filmID, boolean dreiD, String[] genre, String[] sprache): ctor +Film(String titel, String beschreibung, String bildLink, String trailerLink, int dauer, int fsk, int filmID, boolean dreiD): ctor +getGenre(): Genre[] send\_mail::Email\_Sender -transferMail(Session session, String senderMail, String receiverEmail, String subject, String body): void +setGenre(String[] genre): void
+getGenreString(): String
+getSprache(): String[]
+setSprache(String[] sprache): void
+getSpracheString(): String
+getTitel(): String
+setTitel(String titel): void
+getBeschreibung(): String
+setBeschreibung(String beschreibung): void -transferMultipartMail(Session session, String senderMail, String receiverEmail, String subject, String body, String filePath): void +sendMail(String recMail, String subject, String emailBody): void +sendMultipartMail(String recMail, String subject, String emailBody, String filePath): void factory::UpdateFactory +checkVorstellungPLZ(Vorstellung v, int plz, ResultSet mockRs): boolean +checkVorstellungPLZ(Vorstellung v, int plz): boolean -getSession(String recMail): Session factory::ReservierungsStornierungFactory +setBildLink(String bildLink): void +createStornierung(int RNR): ReservierungsStornierung
+getReservierungsStornierungByStrnNR(int StrnNR): ReservierungsStornierung +getTrailerLink(): String +setTrailerLink(String trailerLink): void +getReservierungsStornierungByRNR(int RNR): ReservierungsStornierung +getDauer(): int +getReservierungsStornierungByKID(int KID): ReservierungsStornierung[] +setDauer(int dauer): void +getFsk(): int +setFsk(int fsk): void +getFilmID(): int +setFilmID(int filmID): void factory::LoginFactory +isDreiD(): boolean +getUserLogin(String email, String passwordHash, ResultSet mockRs): UserLogin +setDreiD(boolean dreiD): voi +getUserLogin(String email, String passwordHash): UserLogin +getOrtByPLZ(int plz): Ort ~genrebezeichnung, deskriptor: String +Genre(int genreID, String genrebezeichnung, String deskriptor): ctor factory::ProfilFactory +getGenreID(): int +setGenreID(int genreID): void +getGenrebezeichnung(): String 🔀 +aendereProfil(String vorname, String nachname, String email, String passwort, String strasse, String hausnummer, int PID): void +setGenrebezeichnung(String genrebezeichnung): void +getDeskriptor(): String
+setDeskriptor(String deskriptor): void factory::BuchungsFactory +createBuchungBeleg(int[] sitzeIDs, int[] preiseVerIDs, String seats, String preisVer, int vorstellungsID, int KNR): int +aetBuchungsbelegByBNR(int BNR): Buchungsbeleg +createBuchungsbelegPDF(int KID, Vorstellung vorstellung, Sitz[] sitze): void
+getBuchungsbelegeByKID(int KID): Buchungsbeleg[]
+createBuchungsPositionen(Connection c, int BNR, Sitz[] sitze, int[] preiseVerIDs): void