Visualize First Klassendiagramm Daria Liakhovets hat Workshop_Thema 5 1..* themaID: int 0..* thema: str Workshop workshopID: int themaID: int findet statt Experience dauer: time erfordert experienceID: list(int) experienceID: int experience: str inHouse Workshop Buchung: f inHouse == True { hat 0.. * adresse = customer_address raumID = NULL Workshopprogramm **Trainer** führt adresse = VisualizeFirst_address terminID: int workshopID: int experienceID: int raumID = booked_room beginn : datetime ende: datetime gebucht für **Creative Room** trainer: list (mitarbeiterID) 0.. 1 inHouse: bool roomID: int adresse: str nPersonen: int raumID: int mitPC: bool pcRaum: bool erstellt/pekommt teilnehmer : list (kundenID) nTeilnehmer: int **Angebot** erstellt angebotID: int workshopInfo: list assoziiert mit 0..* preis: float 0.. * bearbeitet Buchung 6 CSC 0..* buchungID : int kundeID : int aufgabenbereich : str Mahnung Privatperson terminID: int angebotID: int mahnungID: int name: int 0... bestätigt: bool rechnungID:int anzahlPersonen: int mahnung: str Kunde zahlungPrüfen(rechnungID): bool hat teilnahmeErfolgt: bool abgeschlossen: bool mahnungErstellen(rechnungID): Mahnung kundeID: int inHouse: bool datum: datetime inhouseBuchung(anfrage): Buchung adresse: str anfrageBeantworten(anfrage, absender): Nachricht workshopBesucht: bool websiteWarten(): Rechnung kundenbeschwerdenKlassifizieren(beschwerde): klasse Unternehmen kundenbeschwerdenWeiterreichen(beschwerde): Nachricht rechnungID: int bonitätsprüfung(kundeID): bool firmenname: int buchungID: int marketing&entwicklung(thema):produkt/aktion preis: float anfrageDatenLöschenWeiterreichen(anfrage):Nachricht findet statt bezahlt: bool assoziiert mit stammdatenwartung() datum: datetime zahlungsverzug: time erstellt erstellt/bekommt Rabattaktion umsatzsteuer: float 0..* kundenadresse: str aktionID: int beinhaltet zahlungstyp: s Nachricht beginn: datetime 0..* skonto: bool ende: datetime prüft nachrichtID: int inhalt: str 0..* Kommunikation kundeID: int 0..* 0..* interaktionID: int email: int kundeID: int text : str manage erstellt/bekommt datum : datetime interaktionTyp: (newsletter, rabattaktion Anfrage_InHouse_Workshop typ: (anfrage, beschwerde,...) inhalt: str 2 datum : datetime kundeID: int anfrageID: int text: str bearbeitet VisThiM adresse: str beinhaltet details: list aufgabenbereich: str 0..1 websiteBetreuung() erstellt Newsletter firmenblogFühren(): newsletterErstellen():Newsletter typ: (allgemein, spezial) content: str empfänger: list(kundenID) **Abteilung** mitarbeiterID: int Mitarbeiter 1 abteilungID: int mitarbeiterID: int stellenbeschreibungID: int 8 name: str (automatisiert) erfasst svNr: int hat abteilungID: int **Arbeitszeit** mitarbeiterID: int Stellenbeschreibung datum: date HR 0..* arbeitstagBeginn: time stelleID: int arbeitstagEnde: time beschreibung: str zuständigkeitsbereich: str MitarbeiterInfo pausenGesamt: time 0..* arbeitszeitGesamt: time erfasst mitarbeiterID: int infoID: int ereignisTyp: (Krankenstand, Urlaub, Fortbildung) beginn: datetime ende: datetime details: str 0..* verwaltet Zusätzliche_Aufgaben Datenschutzbeauftragte mitarbeiterID: int beschreibung: str datenauszugBereitstellen(kundeID, daten): Datenauszug datenLöschen(kundeID): bool