

# Podstawy baz danych

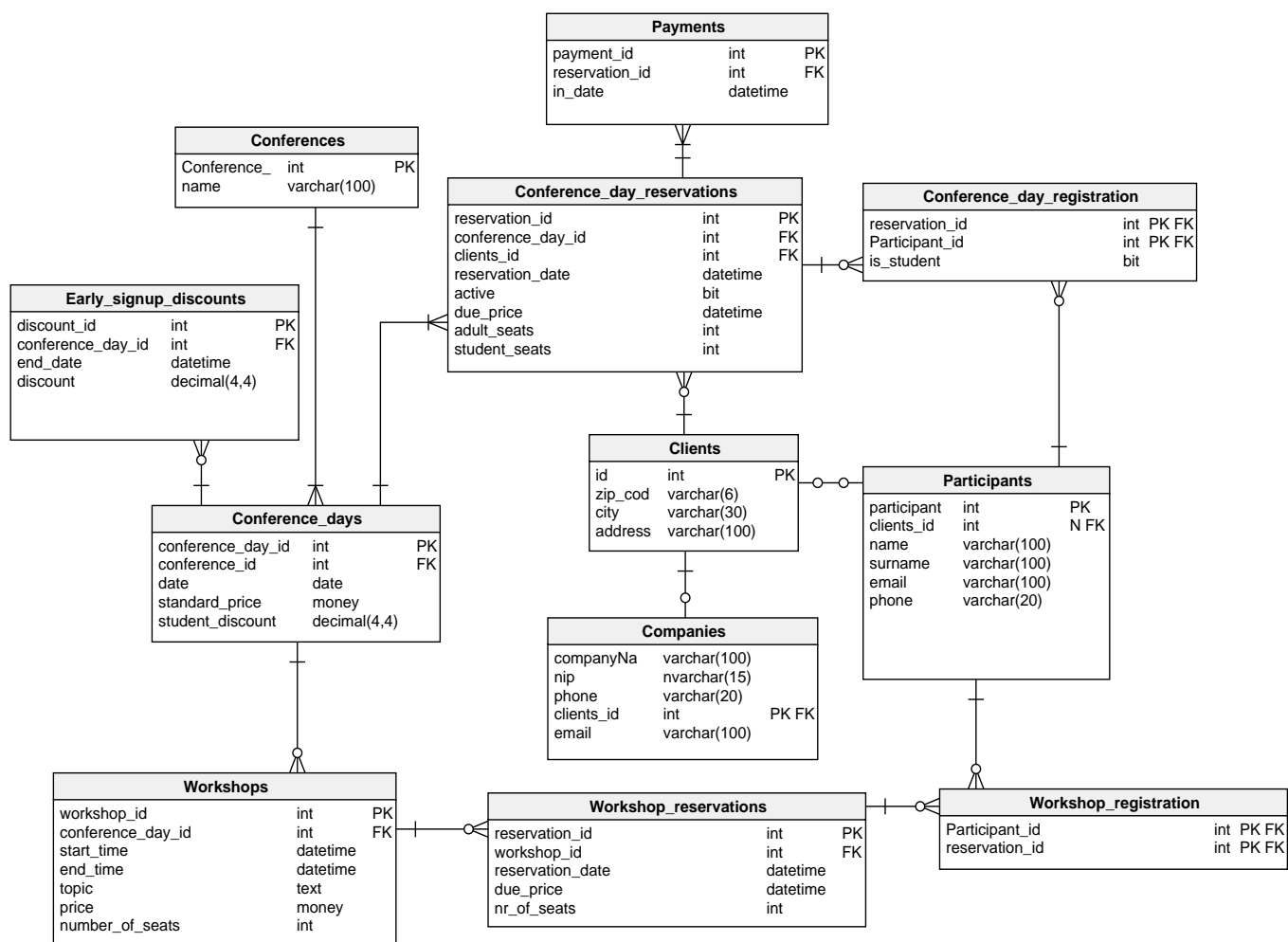
## Projekt konferencje

Agnieszka Dutka, Maciek Trątnowiecki

AGH, Styczeń 2020

### Objaśnienie schematu bazy

- Clients - Reprezentuje klientów chcących opłacić miejsca na konferencjach i warsztatach. Klientem może być zarówno firma, jak i osoba prywatna. W zależności od tego dane klienta reprezentowane są przez odpowiednią relację w bazie.
- Companies - Jeśli klient jest firmą, przechowuje jego dane.
- Participants - Jeśli klient jest osobą prywatną, przechowuje jego dane.
- Conferences - Reprezentuje konferencję z którą powiązane są odpowiednie dni konferencyjne, oraz warsztaty.
- Conference\_days - Reprezentuje pojedynczy dzień konferencji. Powiązana jest z nim ustalona opłata za uczestnictwo. Zniżki obowiązujące w zależności od daty rejestracji zawarte są w relacji Early\_Signup\_Discounts.
- Early\_Signup\_Discounts - Odpowiada za informację o tabeli zniżek na dany dzień konferencyjny. Pojedyncza zniżka przechowywana jest w krotce z atrybutami w postaci procentowej obniżki ceny standardowej, oraz ostatniego dnia w którym obowiązuje.
- Conference\_day\_reservations - Realizuje rezerwacje na poszczególny dzień konferencji. Każda rezerwacja powiązana jest z klientem, który ją opłaca. Za powiązanie rezerwacji z uczestnikiem odpowiada osobna relacja. Zawiera także pole due\_price określające termin płatności. Atrybut active odpowiada za możliwość rezygnacji z podjętej rezerwacji (uznaliśmy, że usuwanie krotki z bazy może nie być optymalnym rozwiązaniem, jako że zawarte w niej dane mogą jeszcze być przydatne z punktu widzenia logiki biznesowej). Atrybuty adult\_seats i student\_seats służą do liczenia kosztu podjęcia rezerwacji przed powiązaniem jej z uczestnikami konferencji.
- Conference\_day\_registration - Wiąże rezerwację z uczestnikami konferencji. Atrybut is\_student informuje, czy danemu uczestnikowi przysługuje zniżka studencka.
- Payments - Przechowuje informacje o wpływach pieniężnych powiązanych z daną rejestracją.
- Workshops - Reprezentuje warsztaty odbywające się w trakcie odpowiednich dni konferencyjnych.
- Workshops\_reservations - Opisuje rezerwacje na warsztaty w sposób analogiczny do rezerwacji na konferencje.
- Workshops\_registrations - Łączy rezerwację z uczestnikami w sposób analogiczny do dni konferencyjnych.



# Implementacja

```
1000 -- tables
1001 -- Table: Clients
1002 CREATE TABLE Clients (
1003     id int NOT NULL IDENTITY,
1004     zip_code varchar(6) NOT NULL,
1005     city varchar(30) NOT NULL,
1006     address varchar(100) NOT NULL,
1007     CONSTRAINT Clients_pk PRIMARY KEY (id)
1008 );
1009
1010 -- Table: Companies
1011 CREATE TABLE Companies (
1012     companyName varchar(100) NOT NULL,
1013     nip nvarchar(15) NOT NULL CHECK ((nip not like '%[0-9]%' ) and (LEN(nip) = 10) and (nip
1014     not like '0%' or nip like '1%')),
1015     phone varchar(20) NOT NULL,
1016     clients_id int NOT NULL,
1017     email varchar(100) NOT NULL CHECK (email like '%_@_%._%'),
1018     CONSTRAINT unique_nip UNIQUE (nip),
1019     CONSTRAINT checkNip CHECK (dbo.IsValidNip(nip) = 1),
1020     CONSTRAINT Companies_pk PRIMARY KEY (clients_id)
1021 );
1022
1023 -- Table: Conference_day_registration
1024 CREATE TABLE Conference_day_registration (
1025     reservation_id int NOT NULL,
1026     Participant_id int NOT NULL,
1027     is_student bit NOT NULL DEFAULT 0,
1028     CONSTRAINT Conference_day_registration_pk PRIMARY KEY (reservation_id ,Participant_id)
1029 );
1030
1031 -- Table: Conference_day_reservations
1032 CREATE TABLE Conference_day_reservations (
1033     reservation_id int NOT NULL IDENTITY,
1034     conference_day_id int NOT NULL,
1035     clients_id int NOT NULL,
1036     reservation_date datetime NOT NULL DEFAULT GETDATE() ,
1037     active bit NOT NULL DEFAULT 1,
1038     due_price datetime NOT NULL DEFAULT DATEADD(week, 2, GETDATE()) CHECK (due_price >=
1039     GETDATE()),
1040     adult_seats int NOT NULL DEFAULT 0 CHECK (adult_seats >= 0),
1041     student_seats int NOT NULL DEFAULT 0 CHECK (student_seats >= 0),
1042     CONSTRAINT Conference_day_reservations_pk PRIMARY KEY (reservation_id)
1043 );
1044
1045 -- Table: Conference_days
1046 CREATE TABLE Conference_days (
1047     conference_day_id int NOT NULL IDENTITY,
1048     conference_id int NOT NULL,
1049     date date NOT NULL DEFAULT GETDATE(),
1050     standard_price money NOT NULL DEFAULT 0 CHECK (standard_price >= 0),
1051     student_discount decimal(4,4) NOT NULL DEFAULT 0 CHECK (student_discount >= 0),
1052     CONSTRAINT Conference_days_pk PRIMARY KEY (conference_day_id)
1053 );
1054
1055 -- Table: Conferences
1056 CREATE TABLE Conferences (
1057     Conference_id int NOT NULL IDENTITY,
1058     name varchar(100) NOT NULL,
1059     CONSTRAINT Conferences_pk PRIMARY KEY (Conference_id)
1060 );
1061
1062 -- Table: Early_signup_discounts
1063 CREATE TABLE Early_signup_discounts (
1064     discount_id int NOT NULL IDENTITY,
1065     conference_day_id int NOT NULL,
1066     end_date datetime NOT NULL,
1067     discount decimal(4,4) NOT NULL DEFAULT 0,
1068     CONSTRAINT Early_signup_discounts_pk PRIMARY KEY (discount_id)
1069 );
1070
1071 -- Table: Participants
1072 CREATE TABLE Participants (
1073     participant_id int NOT NULL IDENTITY,
```

```

1072     clients_id int NULL DEFAULT Null,
1073     name varchar(100) NOT NULL,
1074     surname varchar(100) NOT NULL,
1075     email varchar(100) NOT NULL CHECK (email like '%_@_--%._.-%'),
1076     phone varchar(20) NOT NULL,
1077     CONSTRAINT Participants_pk PRIMARY KEY (participant_id)
1078 );
1079
1080 — Table: Payments
1081 CREATE TABLE Payments (
1082     payment_id int NOT NULL IDENTITY,
1083     reservation_id int NOT NULL,
1084     in_date datetime NOT NULL,
1085     value money NOT NULL,
1086     CONSTRAINT Payments_pk PRIMARY KEY (payment_id)
1087 );
1088
1089 — Table: Workshop_registration
1090 CREATE TABLE Workshop_registration (
1091     Participant_id int NOT NULL,
1092     reservation_id int NOT NULL,
1093     CONSTRAINT Workshop_registration_pk PRIMARY KEY (Participant_id, reservation_id)
1094 );
1095
1096 — Table: Workshop_reservations
1097 CREATE TABLE Workshop_reservations (
1098     reservation_id int NOT NULL IDENTITY,
1099     workshop_id int NOT NULL,
1100     reservation_date datetime NOT NULL DEFAULT GETDATE(),
1101     due_price datetime NOT NULL DEFAULT DATEADD(week, 2, GETDATE()) CHECK (due_price >=
1102     GETDATE()),
1103     nr_of_seats int NOT NULL DEFAULT 0 CHECK (nr_of_seats >= 0),
1104     CONSTRAINT Workshop_reservations_pk PRIMARY KEY (reservation_id)
1105 );
1106
1107 — Table: Workshops
1108 CREATE TABLE Workshops (
1109     workshop_id int NOT NULL IDENTITY,
1110     conference_day_id int NOT NULL,
1111     start_time datetime NOT NULL,
1112     end_time datetime NOT NULL CHECK (end_time >= GETDATE()),
1113     topic text NOT NULL,
1114     price money NOT NULL CHECK (price >= 0),
1115     number_of_seats int NOT NULL DEFAULT 0 CHECK (number_of_seats >= 0),
1116     CONSTRAINT Workshops_pk PRIMARY KEY (workshop_id)
1117 );
1118
1119 — foreign keys
1120 — Reference: Companies_Clients (table: Companies)
1121 ALTER TABLE Companies ADD CONSTRAINT Companies_Clients
1122     FOREIGN KEY (clients_id)
1123     REFERENCES Clients (id);
1124
1125 — Reference: Conference_day_registration_Conference_day_reservations (table:
1126     Conference_day_registration)
1127 ALTER TABLE Conference_day_registration ADD CONSTRAINT
1128     Conference_day_registration_Conference_day_reservations
1129     FOREIGN KEY (reservation_id)
1130     REFERENCES Conference_day_reservations (reservation_id);
1131
1132 — Reference: Conference_day_registration_Participants (table: Conference_day_registration)
1133 ALTER TABLE Conference_day_registration ADD CONSTRAINT
1134     Conference_day_registration_Participants
1135     FOREIGN KEY (Participant_id)
1136     REFERENCES Participants (participant_id);
1137
1138 — Reference: Conference_day_reservations_Clients (table: Conference_day_reservations)
1139 ALTER TABLE Conference_day_reservations ADD CONSTRAINT Conference_day_reservations_Clients
1140     FOREIGN KEY (clients_id)
1141     REFERENCES Clients (id);
1142
1143 — Reference: Conference_day_reservations_Conference_days (table: Conference_day_reservations)
1144 ALTER TABLE Conference_day_reservations ADD CONSTRAINT
1145     Conference_day_reservations_Conference_days
1146     FOREIGN KEY (conference_day_id)
1147     REFERENCES Conference_days (conference_day_id);

```

```

1144 — Reference: Conference_days_Conferences (table: Conference_days)
1145 ALTER TABLE Conference_days ADD CONSTRAINT Conference_days_Conferences
1146 FOREIGN KEY (conference_id)
1147 REFERENCES Conferences (Conference_id);
1148
1149 — Reference: Discounts_Conference_days (table: Early_signup_discounts)
1150 ALTER TABLE Early_signup_discounts ADD CONSTRAINT Discounts_Conference_days
1151 FOREIGN KEY (conference_day_id)
1152 REFERENCES Conference_days (conference_day_id);
1153
1154 — Reference: Participants_Clients (table: Participants)
1155 ALTER TABLE Participants ADD CONSTRAINT Participants_Clients
1156 FOREIGN KEY (clients_id)
1157 REFERENCES Clients (id);
1158
1159 — Reference: Payments_Conference_day_reservations (table: Payments)
1160 ALTER TABLE Payments ADD CONSTRAINT Payments_Conference_day_reservations
1161 FOREIGN KEY (reservation_id)
1162 REFERENCES Conference_day_reservations (reservation_id);
1163
1164 — Reference: Workshop_registration_Participants (table: Workshop_registration)
1165 ALTER TABLE Workshop_registration ADD CONSTRAINT Workshop_registration_Participants
1166 FOREIGN KEY (Participant_id)
1167 REFERENCES Participants (participant_id);
1168
1169 — Reference: Workshop_registration_Workshop_reservations (table: Workshop_registration)
1170 ALTER TABLE Workshop_registration ADD CONSTRAINT Workshop_registration_Workshop_reservations
1171 FOREIGN KEY (reservation_id)
1172 REFERENCES Workshop_reservations (reservation_id);
1173
1174 — Reference: Workshop_reservations_Workshops (table: Workshop_reservations)
1175 ALTER TABLE Workshop_reservations ADD CONSTRAINT Workshop_reservations_Workshops
1176 FOREIGN KEY (workshop_id)
1177 REFERENCES Workshops (workshop_id);
1178
1179 — Reference: Workshops_Conference_days (table: Workshops)
1180 ALTER TABLE Workshops ADD CONSTRAINT Workshops_Conference_days
1181 FOREIGN KEY (conference_day_id)
1182 REFERENCES Conference_days (conference_day_id);
1183
1184 — End of file .

```

../Create.sql

```

1000 CREATE FUNCTION IsValidNip
1001 (
1002     @nip nvarchar(15)
1003 )
1004 RETURNS bit
1005 AS
1006 BEGIN
1007     IF ISNUMERIC(@nip) = 0 BEGIN
1008         RETURN 0
1009     END
1010
1011     IF @nip='0000000000' BEGIN
1012         RETURN 0
1013     END
1014     IF @nip='1234567891' BEGIN
1015         RETURN 0
1016     END
1017     IF @nip='1111111111' BEGIN
1018         RETURN 0
1019     END
1020     IF @nip='1111111112' BEGIN
1021         RETURN 0
1022     END
1023     IF @nip='9999999999' BEGIN
1024         RETURN 0
1025     END
1026     IF @nip='1111111112' BEGIN
1027         RETURN 0
1028     END
1029
1030     DECLARE @sum INT;
1031     SET @sum = 6 * CONVERT(INT, SUBSTRING(@nip,1,1)) +
1032         5 * CONVERT(INT, SUBSTRING(@nip,2,1)) +
1033         7 * CONVERT(INT, SUBSTRING(@nip,3,1)) +
1034         2 * CONVERT(INT, SUBSTRING(@nip,4,1)) +
1035         3 * CONVERT(INT, SUBSTRING(@nip,5,1)) +
1036         4 * CONVERT(INT, SUBSTRING(@nip,6,1)) +
1037         5 * CONVERT(INT, SUBSTRING(@nip,7,1)) +
1038         6 * CONVERT(INT, SUBSTRING(@nip,8,1)) +
1039         7 * CONVERT(INT, SUBSTRING(@nip,9,1));
1040
1041     IF CONVERT(TINYINT, SUBSTRING(@nip,10,1)) = (@sum % 11) BEGIN
1042         RETURN 1
1043     END
1044     RETURN 0
1045 END
1046
1047 -- drop function IsValidNip

```

../checkNip.sql