

# **Sprawozdanie z zadania projektowego**

**Przedmiot:** *Systemy baz danych*

*Obiektowe bazy danych*

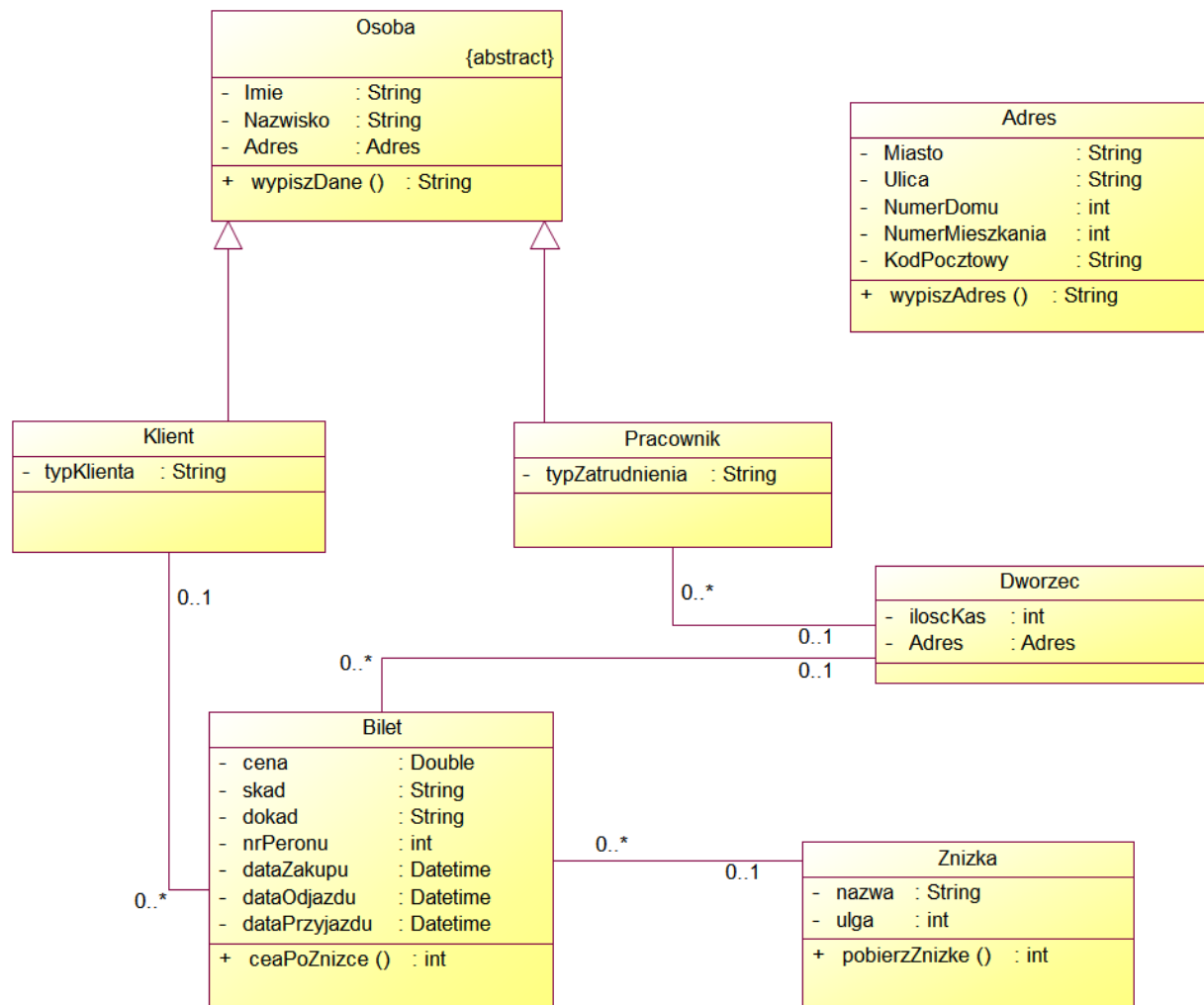
**Grupa:** I7B4S4

**Zespół:** Piechota Katarzyna, Warcholak Anna, Redkiewicz Damian

**Data zajęć:** 2018-04-16

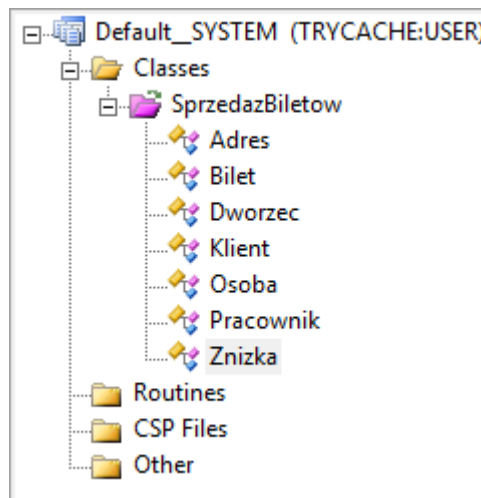
## Identyfikacja klas, metod, związków między klasami dla wybranego obszaru dziedzinowego

Tematem zadania projektowego jest sprzedaż biletów na dworcach kolejowych. W ramach wybranego obszaru dziedzinowego wyróżniono 7 klas. Atrybuty, metody oraz związki pomiędzy poszczególnymi klasami przedstawiono na poniższym diagramie klas.



Rysunek 1. Diagram klas.

## Implementacja obiektowej bazy danych z wykorzystaniem Caché



Rysunek 2. Struktura projektu.

```
Class SprzedazBiletow.Adres Extends (%Persistent, %Populate) [ SqlTableName = Adres ]
{
    Property Miasto As %String(VALUELIST = ",Warszawa,Krakow,Poznan,Bydgoszcz") [ Required ];
    Property Ulica As %String(VALUELIST = ",Akacjowa,Zimowa,Wisniowa,Godowska") [ Required ];
    Property NumerDomu As %Integer(VALUELIST = ",1,2,3") [ Required ];
    Property NumerMieszkania As %Integer(VALUELIST = ",10,20,30") [ Required ];
    Property KodPocztowy As %String(VALUELIST = ",26-689,21-564,65-353") [ Required ];
}
```

Rysunek 3. Klasa Adres.

```
Method wypiszAdres() As %String
{
    Set wynik= ..Ulica_ " " ..NumerDomu_ " " ..NumerMieszkania_ " " ..Miasto_ " " ..KodPocztowy
    Return wynik
}
```

Rysunek 4. Metoda klasy Adres wypiszAdres().

```

Class SprzedazBiletow.Bilet Extends (%Persistent, %Populate) [ SqlTableName = Bilet ]
{
Relationship posiada As SprzedazBiletow.Znizka [ Cardinality = one, Inverse = jestNadana ];
Relationship jestSprzedany As SprzedazBiletow.Dworzec [ Cardinality = one, Inverse = sprzedal ];
Relationship jestKupiony As SprzedazBiletow.Klient [ Cardinality = one, Inverse = zakupil ];
Property cena As %Integer(VALUELIST = ",200,250,300");
Property skad As %String(VALUELIST = ",Warszawa,Kraków,Gdańsk,Katowice");
Property dokad As %String(VALUELIST = ",Wrocław,Lublin,Białystok,Szczecin");
Property nrPeronu As %Integer(VALUELIST = ",1,2,3,4");
Property dataZakupu As %DateTime(VALUELIST = ",2018-05-10 18:38:00 ,2018-04-15 18:20:00 ,2018-05-03 16:15:00");
Property dataOdjazdu As %DateTime(VALUELIST = ",2018-05-15 11:35:00 ,2018-05-15 10:26:00 ,2018-05-15 13:17:00");
Property dataPrzyjazdu As %DateTime(VALUELIST = ",2018-05-15 18:38:00 ,2018-05-15 18:20:00 ,2018-05-15 16:15:00");
}

```

*Rysunek 5. Klasa Bilet.*

```

Method cenaPoZnizce() As %Integer
{
    Set znizka= ..posiada.pobierzZnizke()
    Set wynik= ..cena-znizka
    Return wynik
}

```

*Rysunek 6. Metoda klasy Bilet cenaPoZnizce().*

---

```

Class SprzedazBiletow.Dworzec Extends (%Persistent, %Populate) [ SqlTableName = Dworzec ]
{
Relationship zatrudnia As SprzedazBiletow.Pracownik [ Cardinality = many, Inverse = jestZatrudniony ];
Relationship sprzedal As SprzedazBiletow.Bilet [ Cardinality = many, Inverse = jestSprzedany ];
Property iloscKas As %Integer(VALUELIST = ",2,4,6,8");
Property Adres As Adres;
}

```

*Rysunek 7. Klasa Dworzec.*

---

```

Class SprzedazBiletow.Klient Extends (%Persistent, %Populate, SprzedazBiletow.Osoba) [ SqlTableName = Klient ]
{
Relationship zakupil As SprzedazBiletow.Bilet [ Cardinality = many, Inverse = jestKupiony ];
Property typKlienta As %String(VALUELIST = ",Stały,Nowy");
}

```

*Rysunek 8. Klasa Klient.*

```

Method wypiszDane() As %String
{
    Set wynik= ..Imie_ " " ..Nazwisko_ " " ..typKlienta
    #; _ " " ..Adres.wypiszAdres()
    Return wynik
}

```

*Rysunek 9. Metoda klasy Klient wypiszDane().*

```

Class SprzedazBiletow.Osoba [ Abstract ]
{
    Property Imie As %String (VALUELIST = ",Anna,Jan,Justyna,Krzysztof");
    Property Nazwisko As %String (VALUELIST = ",Nowak,Mazurek,Stefanczyk,Kowal");
    Property Adres As Adres;
}

```

*Rysunek 10. Klasa Osoba.*

```

Class SprzedazBiletow.Pracownik Extends (%Persistent, SprzedazBiletow.Osoba, %Populate) [ SqlTableName = Pracownik ]
{
    Relationship jestZatrudniony As SprzedazBiletow.Dworzec [ Cardinality = one, Inverse = zatrudnia ];
    Property typZatrudnienia As %String (VALUELIST = ",UmowaOPrace,UmowaZlecenie,UmowaODzieło");
}

```

*Rysunek 11. Klasa Pracownik.*

```

Method wypiszDane() As %String
{
    Set wynik= ..Imie_ " " ..Nazwisko_ " " ..typZatrudnienia
    #; _ " " ..Adres.wypiszAdres()
    Return wynik
}

```

*Rysunek 12. Metoda klasy Pracownik wypiszDane().*

```

Class SprzedazBiletow.Znizka Extends (%Persistent, %Populate) [ SqlTableName = Znizka ]
{
    Relationship jestNadana As SprzedazBiletow.Bilet [ Cardinality = many, Inverse = posiada ];
    Property nazwa As %String (VALUELIST = ",Studencka,Senior,RodzinaPlus");
    Property ulga As %Integer (VALUELIST = ",20,50,100");
}

```

*Rysunek 13. Klasa Znizka.*

```

    Method pobierzZnizke() As %Integer
    {
        return ..ulga
    }

```

Rysunek 14. Metoda klasy Znizka pobierzZnizke().

Poniżej przedstawiono wyeksportowaną bazę danych do formatu XML.

```

<?xml version="1.0" encoding="UTF-8"?>
<Export generator="Cache" version="25" zv="Cache for Windows (x86-64) 2017.1.1 (Build
111U_SU)" ts="2018-04-22 10:18:49">
  <Class name="SprzedazBiletow.Adres">
    <SqlTableName>Adres</SqlTableName>
    <Super>%Persistent,%Populate</Super>
    <TimeChanged>64760,35203.853925</TimeChanged>
    <TimeCreated>64759,56465.42616</TimeCreated>

    <Property name="Miasto">
      <Type>%String</Type>
      <Required>1</Required>
      <Parameter name="VALUELIST" value="Warszawa,Krakow,Poznan,Bydgoszcz"/>
    </Property>

    <Property name="Ulica">
      <Type>%String</Type>
      <Required>1</Required>
      <Parameter name="VALUELIST" value="Akacyjowa,Zimowa,Wisniowa,Godowska"/>
    </Property>

    <Property name="NumerDomu">
      <Type>%Integer</Type>
      <Required>1</Required>
      <Parameter name="VALUELIST" value="1,2,3"/>
    </Property>

    <Property name="NumerMieszkania">
      <Type>%Integer</Type>
      <Required>1</Required>
      <Parameter name="VALUELIST" value="10,20,30"/>
    </Property>

    <Property name="KodPocztowy">
      <Type>%String</Type>
      <Required>1</Required>
      <Parameter name="VALUELIST" value="26-689,21-564,65-353"/>
    </Property>

    <Storage name="Default">
      <Type>%Library.CacheStorage</Type>
      <DataLocation>^SprzedazBiletow.AdresD</DataLocation>
      <DefaultData>AdresDefaultData</DefaultData>
      <IdLocation>^SprzedazBiletow.AdresD</IdLocation>
      <IndexLocation>^SprzedazBiletow.AdresI</IndexLocation>
      <StreamLocation>^SprzedazBiletow.AdresS</StreamLocation>
      <ExtentSize>12</ExtentSize>
      <Data name="AdresDefaultData">
        <Value name="1">
          <Value>%CLASSNAME</Value>
        </Value>
        <Value name="2">
          <Value>Miasto</Value>
        </Value>
        <Value name="3">
          <Value>Ulica</Value>
        </Value>
      </Data>
    </Storage>
  </Class>
</Export>

```

```

        <Value name="4">
            <Value>NumerDomu</Value>
        </Value>
        <Value name="5">
            <Value>NumerMieszkania</Value>
        </Value>
        <Value name="6">
            <Value>KodPocztowy</Value>
        </Value>
    </Data>
    <Property name="%%CLASSNAME">
        <Selectivity>100.0000%</Selectivity>
        <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="%%ID">
        <Selectivity>1</Selectivity>
        <AverageFieldSize>1.25</AverageFieldSize>
    </Property>
    <Property name="KodPocztowy">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>6</AverageFieldSize>
    </Property>
    <Property name="Miasto">
        <Selectivity>25.0000%</Selectivity>
        <AverageFieldSize>7.42</AverageFieldSize>
    </Property>
    <Property name="NumerDomu">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="NumerMieszkania">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="Ulica">
        <Selectivity>25.0000%</Selectivity>
        <AverageFieldSize>7.17</AverageFieldSize>
    </Property>
    <SQLMap name="IDKEY">
        <BlockCount>-4</BlockCount>
    </SQLMap>
</Storage>

    <Method name="wypiszAdres">
        <ReturnType>%String</ReturnType>
        <Implementation><![CDATA[
Set wynik= ..Ulica_ " _..NumerDomu_ " _..NumerMieszkania_ " _..Miasto_ " _..KodPocztowy
Return wynik
]]></Implementation>
    </Method>
</Class>

<Class name="SprzedazBiletow.Bilet">
    <SqlTableName>Bilet</SqlTableName>
    <Super>%Persistent,%Populate</Super>
    <TimeChanged>64760,35532.761149</TimeChanged>
    <TimeCreated>64759,76912.480698</TimeCreated>

    <Property name="posiada">
        <Type>SprzedazBiletow.Znizka</Type>
        <Cardinality>one</Cardinality>
        <Inverse>jestNadana</Inverse>
        <Relationship>1</Relationship>
    </Property>

    <Property name="jestSprzedany">
        <Type>SprzedazBiletow.Dworzec</Type>
        <Cardinality>one</Cardinality>
        <Inverse>sprzedal</Inverse>
        <Relationship>1</Relationship>
    </Property>

    <Property name="jestKupiony">
        <Type>SprzedazBiletow.Klient</Type>
        <Cardinality>one</Cardinality>
        <Inverse>zakupil</Inverse>
    </Property>

```

```

    <Relationship>1</Relationship>
  </Property>

  <Property name="cena">
    <Type>%Integer</Type>
    <Parameter name="VALUELIST" value=" ,200,250,300"/>
  </Property>

  <Property name="skad">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value=" ,Warszawa,Kraków,Gdańsk,Katowice"/>
  </Property>

  <Property name="dokad">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value=" ,Wrocław,Lublin,Białystok,Szczecin"/>
  </Property>

  <Property name="nrPeronu">
    <Type>%Integer</Type>
    <Parameter name="VALUELIST" value=" ,1,2,3,4"/>
  </Property>

  <Property name="dataZakupu">
    <Type>%DateTime</Type>
    <Parameter name="VALUELIST" value=" ,2018-05-10 18:38:00 ,2018-04-15 18:20:00
,2018-05-03 16:15:00"/>
  </Property>

  <Property name="dataOdjazdu">
    <Type>%DateTime</Type>
    <Parameter name="VALUELIST" value=" ,2018-05-15 11:35:00 ,2018-05-15 10:26:00
,2018-05-15 13:17:00"/>
  </Property>

  <Property name="dataPrzyjazdu">
    <Type>%DateTime</Type>
    <Parameter name="VALUELIST" value=" ,2018-05-15 18:38:00 ,2018-05-15 18:20:00
,2018-05-15 16:15:00"/>
  </Property>

  <Method name="cenaPoZnizce">
    <ReturnType>%Integer</ReturnType>
    <Implementation><![CDATA[
Set znizka= ..posiada.pobierzZnizke()
Set wynik= ..cena-znizka
Return wynik
]]></Implementation>
  </Method>

  <Storage name="Default">
    <Type>%Library.CacheStorage</Type>
    <DataLocation>^SprzedazBiletow.BiletD</DataLocation>
    <DefaultData>BiletDefaultData</DefaultData>
    <IdLocation>^SprzedazBiletow.BiletD</IdLocation>
    <IndexLocation>^SprzedazBiletow.BiletI</IndexLocation>
    <StreamLocation>^SprzedazBiletow.BiletS</StreamLocation>
    <ExtentSize>5</ExtentSize>
    <Data name="BiletDefaultData">
      <Value name="1">
        <Value>%CLASSNAME</Value>
      </Value>
      <Value name="2">
        <Value>posiada</Value>
      </Value>
      <Value name="3">
        <Value>jestSprzedany</Value>
      </Value>
      <Value name="4">
        <Value>jestKupiony</Value>
      </Value>
      <Value name="5">
        <Value>cenaZakupu</Value>
      </Value>
      <Value name="6">
        <Value>skad</Value>
      </Value>
    </Data>
  </Storage>

```



```

        <Value name="7">
            <Value>dokad</Value>
        </Value>
        <Value name="8">
            <Value>nrPeronu</Value>
        </Value>
        <Value name="9">
            <Value>dataZakupu</Value>
        </Value>
        <Value name="10">
            <Value>dataOdjazdu</Value>
        </Value>
        <Value name="11">
            <Value>dataPrzyjazdu</Value>
        </Value>
        <Value name="12">
            <Value>cena</Value>
        </Value>
    </Data>
    <Property name="%%CLASSNAME">
        <Selectivity>100.0000%</Selectivity>
        <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="%%ID">
        <Selectivity>1</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="cena">
        <Selectivity>50.0000%</Selectivity>
        <AverageFieldSize>3</AverageFieldSize>
    </Property>
    <Property name="dataOdjazdu">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>19</AverageFieldSize>
    </Property>
    <Property name="dataPrzyjazdu">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>19</AverageFieldSize>
    </Property>
    <Property name="dataZakupu">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>19</AverageFieldSize>
    </Property>
    <Property name="dokad">
        <Selectivity>50.0000%</Selectivity>
        <AverageFieldSize>7.2</AverageFieldSize>
    </Property>
    <Property name="jestKupiony">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="jestSprzedany">
        <Selectivity>100.0000%</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="nrPeronu">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="posiada">
        <Selectivity>20.0000%</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="skad">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>7.2</AverageFieldSize>
    </Property>
    <SQLMap name="IDKEY">
        <BlockCount>-4</BlockCount>
    </SQLMap>
</Storage>
</Class>

<Class name="SprzedazBiletow.Dworzec">
    <SqlTableName>Dworzec</SqlTableName>
    <Super>%Persistent,%Populate</Super>

```

```

<TimeChanged>64760,35516.367712</TimeChanged>
<TimeCreated>64759,75594.621704</TimeCreated>

<Property name="zatrudnia">
  <Type>SprzedazBiletow.Pracownik</Type>
  <Cardinality>many</Cardinality>
  <Inverse>jestZatrudniony</Inverse>
  <Relationship>1</Relationship>
</Property>

<Property name="sprzedal">
  <Type>SprzedazBiletow.Bilet</Type>
  <Cardinality>many</Cardinality>
  <Inverse>jestSprzedany</Inverse>
  <Relationship>1</Relationship>
</Property>

<Property name="iloscKas">
  <Type>%Integer</Type>
  <Parameter name="VALUELIST" value=",2,4,6,8"/>
</Property>

<Property name="Adres">
  <Type>Adres</Type>
</Property>

<Storage name="Default">
  <Type>%Library.CacheStorage</Type>
  <DataLocation>^SprzedazBiletow.DworzecD</DataLocation>
  <DefaultData>DworzecDefaultData</DefaultData>
  <IdLocation>^SprzedazBiletow.DworzecD</IdLocation>
  <IndexLocation>^SprzedazBiletow.DworzecI</IndexLocation>
  <StreamLocation>^SprzedazBiletow.DworzecS</StreamLocation>
  <ExtentSize>5</ExtentSize>
  <Data name="DworzecDefaultData">
    <Value name="1">
      <Value>%CLASSNAME</Value>
    </Value>
    <Value name="2">
      <Value>iloscKas</Value>
    </Value>
    <Value name="3">
      <Value>Adres</Value>
    </Value>
  </Data>
  <Property name="%CLASSNAME">
    <Selectivity>100.0000</Selectivity>
    <AverageFieldSize>1</AverageFieldSize>
  </Property>
  <Property name="%ID">
    <Selectivity>1</Selectivity>
    <AverageFieldSize>2</AverageFieldSize>
  </Property>
  <Property name="Adres">
    <Selectivity>33.3333</Selectivity>
    <AverageFieldSize>1.8</AverageFieldSize>
  </Property>
  <Property name="iloscKas">
    <Selectivity>33.3333</Selectivity>
    <AverageFieldSize>1</AverageFieldSize>
  </Property>
  <SQLMap name="IDKEY">
    <BlockCount>-4</BlockCount>
  </SQLMap>
</Storage>
</Class>

<Class name="SprzedazBiletow.Klient">
  <SqlTableName>Klient</SqlTableName>
  <Super>%Persistent,%Populate,SprzedazBiletow.Osoba</Super>
  <TimeChanged>64760,35528.44182</TimeChanged>
  <TimeCreated>64759,65301.553514</TimeCreated>

  <Property name="zakupil">
    <Type>SprzedazBiletow.Bilet</Type>
    <Cardinality>many</Cardinality>

```

```

    <Inverse>jestKupiony</Inverse>
    <Relationship>1</Relationship>
  </Property>

  <Property name="typKlienta">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value=",Stały,Nowy"/>
  </Property>

  <Method name="wypiszDane">
    <ReturnType>%String</ReturnType>
    <Implementation><![CDATA[
Set wynik= ..Imie_ " " ..Nazwisko_ " " ..typKlienta
#; " " ..Adres.wypiszAdres()
Return wynik
]]></Implementation>
  </Method>

  <Storage name="Default">
    <Type>%Library.CacheStorage</Type>
    <DataLocation>^SprzedazBiletow.KlientD</DataLocation>
    <DefaultData>KlientDefaultData</DefaultData>
    <IdLocation>^SprzedazBiletow.KlientD</IdLocation>
    <IndexLocation>^SprzedazBiletow.KlientI</IndexLocation>
    <StreamLocation>^SprzedazBiletow.Klients</StreamLocation>
    <ExtentSize>5</ExtentSize>
    <Data name="KlientDefaultData">
      <Value name="1">
        <Value>%CLASSNAME</Value>
      </Value>
      <Value name="2">
        <Value>Imie</Value>
      </Value>
      <Value name="3">
        <Value>Nazwisko</Value>
      </Value>
      <Value name="4">
        <Value>Adres</Value>
      </Value>
      <Value name="5">
        <Value>typKlienta</Value>
      </Value>
      <Value name="6">
        <Value>test1</Value>
      </Value>
    </Data>
    <Property name="%%CLASSNAME">
      <Selectivity>100.0000</Selectivity>
      <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="%%ID">
      <Selectivity>1</Selectivity>
      <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="Adres">
      <Selectivity>33.3333</Selectivity>
      <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="Imie">
      <Selectivity>33.3333</Selectivity>
      <AverageFieldSize>6.6</AverageFieldSize>
    </Property>
    <Property name="Nazwisko">
      <Selectivity>33.3333</Selectivity>
      <AverageFieldSize>7.8</AverageFieldSize>
    </Property>
    <Property name="typKlienta">
      <Selectivity>50.0000</Selectivity>
      <AverageFieldSize>4.4</AverageFieldSize>
    </Property>
    <SQLMap name="IDKEY">
      <BlockCount>-4</BlockCount>
    </SQLMap>
  </Storage>
</Class>

```

```

<Class name="SprzedazBiletow.Osoba">
  <Abstract>1</Abstract>
  <TimeChanged>64760,29818.006541</TimeChanged>
  <TimeCreated>64759,65224.04298</TimeCreated>

  <Property name="Imie">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value="Anna,Jan,Justyna,Krzysztof"/>
  </Property>

  <Property name="Nazwisko">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value="Nowak,Mazurek,Stefanczyk,Kowal"/>
  </Property>

  <Property name="Adres">
    <Type>Adres</Type>
  </Property>

  <Method name="wypiszDane">
    <ReturnType>%String</ReturnType>
  </Method>
</Class>

<Class name="SprzedazBiletow.Pracownik">
  <SqlTableName>Pracownik</SqlTableName>
  <Super>%Persistent,SprzedazBiletow.Osoba,%Populate</Super>
  <TimeChanged>64760,35522.57064</TimeChanged>
  <TimeCreated>64759,73949.039324</TimeCreated>

  <Property name="jestZatrudniony">
    <Type>SprzedazBiletow.Dworzec</Type>
    <Cardinality>one</Cardinality>
    <Inverse>zatrudnia</Inverse>
    <Relationship>1</Relationship>
  </Property>

  <Property name="typZatrudnienia">
    <Type>%String</Type>
    <Parameter name="VALUELIST" value="UmowaOPrace,UmowaZlecenie,UmowaODzieło"/>
  </Property>

  <Method name="wypiszDane">
    <ReturnType>%String</ReturnType>
    <Implementation><![CDATA[
Set wynik= ..Imie_"_"..Nazwisko_"_"..typZatrudnienia
#; _"_"..Adres.wypiszAdres()
Return wynik
]]></Implementation>
  </Method>

  <Storage name="Default">
    <Type>%Library.CacheStorage</Type>
    <DataLocation>^SprzedazBiletow.PracownikD</DataLocation>
    <DefaultData>PracownikDefaultData</DefaultData>
    <IdLocation>^SprzedazBiletow.PracownikD</IdLocation>
    <IndexLocation>^SprzedazBiletow.PracownikI</IndexLocation>
    <StreamLocation>^SprzedazBiletow.PracownikS</StreamLocation>
    <ExtentSize>5</ExtentSize>
    <Data name="PracownikDefaultData">
      <Value name="1">
        <Value>%CLASSNAME</Value>
      </Value>
      <Value name="2">
        <Value>Imie</Value>
      </Value>
      <Value name="3">
        <Value>Nazwisko</Value>
      </Value>
      <Value name="4">
        <Value>Adres</Value>
      </Value>
      <Value name="5">
        <Value>typZatrudnienia</Value>
      </Value>
    </Data>
  </Storage>
</Class>

```

```

        </Value>
        <Value name="6">
            <Value>jestZatrudniony</Value>
        </Value>
    </Data>
    <Property name="%%CLASSNAME">
        <Selectivity>100.0000%</Selectivity>
        <AverageFieldSize>1</AverageFieldSize>
    </Property>
    <Property name="%%ID">
        <Selectivity>1</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="Adres">
        <Selectivity>20.0000%</Selectivity>
        <AverageFieldSize>1.4</AverageFieldSize>
    </Property>
    <Property name="Imie">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>4.8</AverageFieldSize>
    </Property>
    <Property name="Nazwisko">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>5.4</AverageFieldSize>
    </Property>
    <Property name="jestZatrudniony">
        <Selectivity>25.0000%</Selectivity>
        <AverageFieldSize>2</AverageFieldSize>
    </Property>
    <Property name="typZatrudnienia">
        <Selectivity>33.3333%</Selectivity>
        <AverageFieldSize>11.8</AverageFieldSize>
    </Property>
    <SQLMap name="IDKEY">
        <BlockCount>-4</BlockCount>
    </SQLMap>
</Storage>
</Class>

<Class name="SprzedazBiletow.Znizka">
    <SqlTableName>Znizka</SqlTableName>
    <Super>%Persistent,%Populate</Super>
    <TimeChanged>64760,35506.032586</TimeChanged>
    <TimeCreated>64759,76844.268579</TimeCreated>

    <Property name="jestNadana">
        <Type>SprzedazBiletow.Bilet</Type>
        <Cardinality>many</Cardinality>
        <Inverse>posiada</Inverse>
        <Relationship>1</Relationship>
    </Property>

    <Property name="nazwa">
        <Type>%String</Type>
        <Parameter name="VALUELIST" value=","Studencka,Senior,RodzinaPlus"/>
    </Property>

    <Property name="ulga">
        <Type>%Integer</Type>
        <Parameter name="VALUELIST" value=","20,50,100"/>
    </Property>

    <Method name="pobierzZnizke">
        <ReturnType>%Integer</ReturnType>
        <Implementation><![CDATA[ return ..ulga
]]></Implementation>
    </Method>

    <Storage name="Default">
        <Type>%Library.CacheStorage</Type>
        <DataLocation>^SprzedazBiletow.ZnizkaD</DataLocation>
        <DefaultData>ZnizkaDefaultData</DefaultData>
        <IdLocation>^SprzedazBiletow.ZnizkaD</IdLocation>
        <IndexLocation>^SprzedazBiletow.ZnizkaI</IndexLocation>
        <StreamLocation>^SprzedazBiletow.ZnizkaS</StreamLocation>
        <ExtentSize>5</ExtentSize>
    </Storage>

```

```

<Data name="ZnizkaDefaultData">
  <Value name="1">
    <Value>%%CLASSNAME</Value>
  </Value>
  <Value name="2">
    <Value>nazwa</Value>
  </Value>
  <Value name="3">
    <Value>ulga</Value>
  </Value>
</Data>
<Property name="%%CLASSNAME">
  <Selectivity>100.0000</Selectivity>
  <AverageFieldSize>1</AverageFieldSize>
</Property>
<Property name="%%ID">
  <Selectivity>1</Selectivity>
  <AverageFieldSize>2</AverageFieldSize>
</Property>
<Property name="nazwa">
  <Selectivity>50.0000</Selectivity>
  <AverageFieldSize>7.2</AverageFieldSize>
</Property>
<Property name="ulga">
  <Selectivity>50.0000</Selectivity>
  <AverageFieldSize>2.6</AverageFieldSize>
</Property>
<SQLMap name="IDKEY">
  <BlockCount>-4</BlockCount>
</SQLMap>
</Storage>
</Class>
</Export>

```

*Listing 1. Schemat bazy danych w XML.*

## Wypełnienie bazy danych testowymi danymi

W Cache można wykorzystać komendę *Populate()* w celu auto-populacji tabeli. Ponadto można skorzystać z instrukcji SQL *INSERT INTO nazwa\_tabeli(...kolumny) VALUES (...wartości)*.

Przykładowe wywołanie komendy *Populate()* dla klasy *Adres*:

*Do Class(SprzedazBiletow.Adres).Populate(12)*

```
USER>>select * from SprzedazBiletow.Adres
1.      select * from SprzedazBiletow.Adres
```

ID	KodPocztowy	Miasto	NumerDomu	NumerMieszkania	Ulica
1	65-353	Warszawa	3	20	Godowska
2	21-564	Krakow 1	10		Wisniowa
3	21-564	Krakow 2	30		Zimowa
4	65-353	Warszawa	2	30	Wisniowa
5	26-689	Bydgoszcz	2	10	Zimowa
6	65-353	Warszawa	2	10	Wisniowa
7	21-564	Poznan 2	30		Akacyjowa
8	21-564	Krakow 2	10		Wisniowa
9	21-564	Warszawa	3	30	Godowska
10	21-564	Bydgoszcz	2	30	Zimowa
11	26-689	Poznan 2	10		Zimowa
12	26-689	Bydgoszcz	3	10	Zimowa

Rysunek 15. Tabela *Adres* z przykładowymi danymi.

```
USER>>select * from SprzedazBiletow.Bilet
13.      select * from SprzedazBiletow.Bilet
```

ID	cena	dataOdjazdu	dataPrzyjazdu	dataZakupu	dokad	jestKupi
ony	jestSprzedany	nrPeronu	posiada	skad		
31	300	2018-05-15 11:35:00	2018-05-15 18:38:00	2018-05-10 18:38		
:00	Lublin	29	12	3	87	Kraków
32	300	2018-05-15 13:17:00	2018-05-15 18:20:00	2018-05-03 16:15		
:00	Lublin	28	12	4	90	Katowice
33	200	2018-05-15 10:26:00	2018-05-15 18:20:00	2018-04-15 18:20		
:00	Szczecin	28	12	3	91	Warszawa
34	200	2018-05-15 11:35:00	2018-05-15 18:20:00	2018-04-15 18:20		
:00	Szczecin	28	12	2	89	Warszawa
35	200	2018-05-15 13:17:00	2018-05-15 16:15:00	2018-04-15 18:20		
:00	Szczecin	27	12	2	88	Kraków

Rysunek 16. Tabela *Bilet* z przykładowymi danymi.

```
USER>>select * from SprzedazBiletow.Dworzec
12.      select * from SprzedazBiletow.Dworzec
```

ID	Adres	iloscKas
11	10	6
12	1	8
13	10	4
14	11	8
15	11	4

Rysunek 17. Tabela Dworzec z przykładowymi danymi.

```
USER>>select * from SprzedazBiletow.Klient
10.      select * from SprzedazBiletow.Klient
```

ID	Adres	Imie	Nazwisko	typKlienta
27	8	Krzysztof	Stefanczyk	Staly
28	5	Justyna	Stefanczyk	Nowy
29	5	Jan	Kowal	Nowy
30	5	Justyna	Mazurek	Nowy
31	9	Justyna	Mazurek	Staly

Rysunek 18. Tabela Klient z przykładowymi danymi.

```
USER>>select * from SprzedazBiletow.Pracownik
11.      select * from SprzedazBiletow.Pracownik
```

ID	Adres	Imie	Nazwisko	jestZatrudniony	typZatrudnienia
11	11	Anna	Nowak	15	UmowaODzieło
12	8	Anna	Kowal	15	UmowaZlecenie
13	1	Anna	Kowal	14	UmowaODzieło
14	2	Jan	Kowal	13	UmowaOPrace
15	10	Krzysztof	Mazurek	12	UmowaOPrace

Rysunek 19. Tabela Pracownik z przykładowymi danymi.

```
USER>>select * from SprzedazBiletow.Znizka
9.      select * from SprzedazBiletow.Znizka
```

ID	nazwa	ulga
87	Senior	50
88	Studencka	100
89	Studencka	50
90	Senior	100
91	Senior	100

Rysunek 20. Tabela Znizka z przykładowymi danymi.



## Wykonanie metod obiektów w terminalu

```
USER>set adres = ##class(SprzedazBiletow.Adres).%OpenId(2)

USER>set wypiszAdres = adres.wypiszAdres()

USER>write wypiszAdres
Wisniowa 1 10 Krakow 21-564
_
```

Rysunek 21. Wywołanie metody wypiszAdres() dla obiektu typu Adres.

```
17.      select b.ID, b.cena, z.ulga from SprzedazBiletow.Bilet as b join SprzedazBiletow.Znizka as z on b.posiada = z.ID
```

ID	cena	ulga
31	300	50
32	300	100
33	200	100
34	200	50
35	200	100

5 Rows(s) Affected

statement prepare time(s)/globals/lines/disk: 0.0961s/46358/297116/0ms

execute time(s)/globals/lines/disk: 0.0003s/11/1002/0ms

cached query class: %sqlcq.USER.cls27

-----  
USER>>q

```
USER>set bilet = ##class(SprzedazBiletow.Bilet).%OpenId(31)
```

```
USER>set cenaPoZnizce = bilet.cenaPoZnizce()
```

```
USER>write cenaPoZnizce
```

250  
\_

Rysunek 22. Wywołanie metody cenaPoZnizce() dla obiektu typu Bilet.

```

USER>>select * from SprzedazBiletow.Klient
18.      select * from SprzedazBiletow.Klient

ID      Adres  Imie      Nazwisko      typKlienta
27      8      Krzysztof  Stefanczyk    Staly
28      5      Justyna   Stefanczyk    Nowy
29      5      Jan       Kowal         Nowy
30      5      Justyna   Mazurek       Nowy
31      9      Justyna   Mazurek       Staly

5 Rows(s) Affected
statement prepare time(s)/globals/lines/disk: 0.0002s/5/175/0ms
execute time(s)/globals/lines/disk: 0.0004s/6/1296/0ms
cached query class: %sqlcq.USER.cls4
-----

```

```

USER>>q

```

```

USER>set klient = ##class(SprzedazBiletow.Klient).%OpenId(27)

```

```

USER>set daneKlienta = klient.wypiszDane()

```

```

USER>write daneKlienta
Krzysztof Stefanczyk Staly

```

*Rysunek 23. Wywołanie metody wypiszDane() dla obiektu typu Klient.*

```

USER>>select * from SprzedazBiletow.Pracownik
20.      select * from SprzedazBiletow.Pracownik

ID      Adres  Imie      Nazwisko      jestZatrudniony typZatrudnienia
11      11      Anna      Nowak         15             UmowaODzieło
12      8      Anna      Kowal         15             UmowaZlecenie
13      1      Anna      Kowal         14             UmowaODzieło
14      2      Jan       Kowal         13             UmowaOPrace
15      10     Krzysztof Mazurek      12             UmowaOPrace

5 Rows(s) Affected
statement prepare time(s)/globals/lines/disk: 0.0001s/5/157/0ms
execute time(s)/globals/lines/disk: 0.0003s/6/1478/0ms
cached query class: %sqlcq.USER.cls18
-----

```

```

USER>>q

```

```

USER>set pracownik = ##class(SprzedazBiletow.Pracownik).%OpenId(14)

```

```

USER>set danePracownik = pracownik.wypiszDane()

```

```

USER>write danePracownik
Jan Kowal UmowaOPrace

```

*Rysunek 24. Wywołanie metody wypiszDane() dla obiektu typu Pracownik.*

## Napotkane problemy i ich rozwiązania

- utrudnione przechodzenie między oknem do składni SQL (w celu wykonywania zapytań), a terminalem (w celu wywołania metod na obiektach) – brak rozwiązania
- nieeleganckie wyświetlanie klasy z dużą ilością atrybutów – brak rozwiązania
- brak możliwości powiększenia okna do pełnej rozdzielczości ekranu – brak rozwiązania
- brak możliwości wskazania w kreatorze stereotypu klasy (*%Persistent*) dla klasy dziedziczącej (*Extends*) – konieczność ręcznego dopisania skryptu
- niedziałająca konkatenacja Stringów przy użyciu operatora „+” – zgodnie z dokumentacją, konkatenacja przebiega za pomocą operatora „\_”

## Pomysły/idee

- stworzenie konsoli umożliwiającej wygodne wykonywanie zapytań SQL oraz wywoływania metod na obiektach,
- dodanie horyzontalnego scrollbara do przeglądania konsoli (np. wyników zapytania)
- możliwość skalowania okna programu do rozmiarów ekranu

## Ocena środowiska

Caché dostarcza wygodnego mechanizmu tworzenia klas przy użyciu kreatora. Znacznie ułatwia to tworzenie bazy danych, lecz nie wszystkie dane można „wyklikać”. Ponadto kreator pozwala łatwo tworzyć relacje między klasami. Silnik dostarcza też użytecznego mechanizmu populacji danych, ale nie wszystkie typy danych da się generować. Dokumentacja programu jest użyteczna - ułatwia poruszanie się w dosyć nieintuicyjnym środowisku.

Obiektowa baza danych zdecydowanie ułatwia integrację z aplikacjami stworzonymi w obiektowym języku programowania, jednakże obiektowe systemy zarządzania bazami danych nie cieszą się wielkim uznaniem. Caché, mimo wielu wad, znajduje się na pierwszym miejscu według rankingu z maja 2018 roku.

## Tabela

Jak środowisko realizuje zagadnienia związane z:	Odpowiedź
Obiektowością - Metodami	Bardzo dobrze
Obiektowością – Dziedziczeniem, typami abstrakcyjnymi	Dostatecznie (nie wszystko da się „wyklikać”)
Obiektowością – Związkami między klasami (asocjacja, kompozycja)	Dostatecznie (nie wszystko da się „wyklikać”)
Obiektowością – Typy danych – proste, złożone	Bardzo dobrze
Obiektowością - Polimorfizmem	Bardzo dobrze
Obiektowością – Tożsamością danych	Bardzo dobrze
Obiektowością – Enkapsulacją	Bardzo dobrze
Obiektowością – Trwałością danych	Bardzo dobrze
Administracja – Zarządzanie środowiskiem	Dostatecznie (konieczność uruchamiania terminala i studia w dwóch różnych oknach)
Interfejs – Czy narzędzie posiada API? Dla jakich języków?	Tak – dla języka SQL
Środowisko – Czy narzędzie zawiera w sobie środowisko programistyczne?	Tak – możliwość wykonywania zapytań sql i metod na obiektach
Skalowalność – Czy narzędzie umożliwia horyzontalne skalowanie środowiska? (rozproszone przetwarzanie, magazynowanie, replikacja)	Tak
Multi-model – Czy narzędzie zapewnia inne rodzaje bazy danych?	Tak (key-value store, Object oriented DBMS, Relational DBMS)

## Adres repozytorium

<https://github.com/KatiePi12/SprzedazBiletow>