

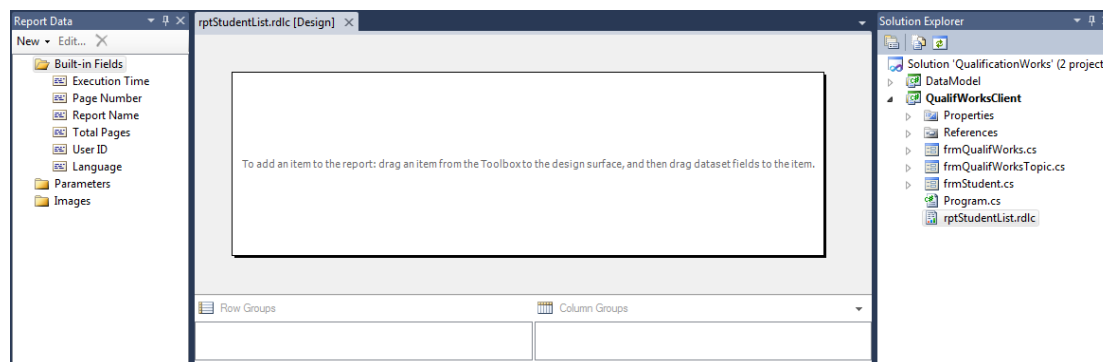
Šī laboratorijas darba mērķis ir iepazīties ar datu avotu sasaistes iespējām ar sagatavotiem pārskatu šabloniem un to atspoguļošanu vizuālajās kontrolēs. Tiek apskatīta pārskatu projektēšana un vizuālie komponenti, kas nodrošina pārskatu sagatavošanu, attēlošanu un izdrukāšanu.

Izstrādes vide piedāvā vizuālu pārskatu projektēšanas rīku (**Report Designer**), kas ļauj sagatavot pārskatus gan no relāciju, gan daudzdimensiju datu avotiem. Iepazīšanos ar šo rīku un tā sniegtajām iespējām, apskatīsim realizējot uzdevumu pasniedzēja vadībā.

6.1 UZDEVUMS PASNIEDZĒJA VADĪBĀ

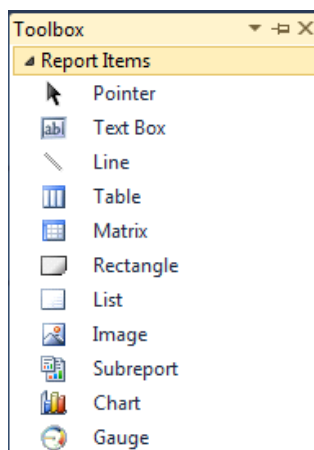
Pārskatu projektēšanas iespējas apskatīsim uz pārskata piemēru, kura uzdevums būs izvadīt datubāzes tabulā **Students** esošos datus tabulas veidā. Lai to izdarītu, projektam **QualifWorksClient** jāpievieno pārskata resurss – uz projekta **QualifWorksClient** virsotnes konteksta izvēlnes jāizvēlēties punktu **Add | New Item...** un dialogā, kas atvērsies, izvēlēties kategorijas **Reporting** sagatavi **Report**. Laukā **Name:** norādīt **rptStudentList.rdlc**.

Nospiežot pogu **Add** pārskata sagatave tiek pievienota projektam un automātiski atvērta pārskatu projektēšanas rīkā (versijai Visual Studio 2010):



Pēc noklusēšanas pārskata sastāv no tukša laukuma – sekcijas **Body**, uz kura no rīku loga **Toolbox** var uzvietot pārskata kontroles. Sekcija **Body** ir paredzēta pārskata galvenes, satura un kājenes datu atspoguļošanai. Papildus sekcija – katras lappuses galvenes (*Header*) un kājenes (*Footer*) datiem ir pievienojama, uz pārskata redaktora tukšās (pelēkās) daļas konteksta izvēlnes izvēloties attiecīgi punktus **Add Page Header** un/vai **Add Page Footer** (vai arī no galvenās izvēlnes **Report** tāda paša nosaukuma apakšpunktiem).

Par redaktora kreisajā pusē esošā loga **Report Data** mērķi un saturu, tiks pastāstīts vēlāk, bet, atverot rīku logu, tiek piedāvāti pārskata vizuālie komponenti, no kuriem var veidot pārskata dizainu:



Izveidosim pārskata lappuses kājenes sekciju, kurā tiks izvadīta informācija par tekošo un kopējo lappušu skaitu:

- Pārskatam **rptStudentList.rdlc** pievienot lappuses kājenes sekciju.
- No rīku loga uz pārskata kājenes sekcijas uzvietot pārskata komponentu **Textbox**, un uzstādīt īpašības:

Īpašība:	Vērtība:
Font	Arial; 8pt;
Location	7cm; 0cm
Size	3cm; 0,5cm
TextAlign	Center
Value	= "Lapa " & Globals!PageNumber & " no " & Globals!TotalPages

Komponenta īpašībā **Value** norādīta izteiksme, kuras mērķis ir katras lappuses kājenes daļā izvadīt informāciju par tekošo lappusi un kopējo lappušu skaitu. Jāņem vērā, ka īpašībā **Value** vērtības jānorāda kā programmēšanas valodas Visual Basic izteiksmes, par virkņu konkatēnācijas operatoru izmantojot ampersanta rakstzīmi &. Kā redzams no lauka **Value** vērtības, tajā tiek izmantoti divi iepriekš definēti grupas **Globals** mainīgie. Pirmais atgriež tekošās lappuses numuru, bet otrais – kopējo lappušu skaitu, kas nepieciešamas, lai pilnībā izvadītu visus pārskata datus.

Pāriesim pie pārskata sekcijas **Body**, un specificēsim pārskatu, kura sākumā tiek izvadīts pārskata virsraksts, aiz kura seko tabulas kolonnu virsraksti un paši tabulas dati, ar šādu gala dizainu:

- Visual Studio 2008:

Studentu saraksts					
Nr.p.k.	Vārds	Uzvārds	Studiju līmenis	Parādi?	Piezīmes
=RowNr	=Fields!Name.V	=Fields!Surname	=Fields!Level.Val	=If(Fields	=Fields!Notes.Value
Footer					
Page Footer					
= "Lapa " & Globals.PageNumber & " no " & Globals.TotalPages					

- Visual Studio 2010:

Studentu saraksts					
Nr.p.k.	Vārds	Uzvārds	Studiju līmenis	Parādi?	Piezīmes
«Expr»	[Name]	[Surname]	[Level]	«Expr»	[Notes]
«Expr»					

Redzams, ka starp izstrādes vides versijām, pārskatu redaktors informāciju attēlo būtiski savādāk.

Bet pirms pārskata elementu izvietojšanas ir jānodrošina pieeja datu avotam, kura rakstu vērtības tiks izmantotas par pamatu pārskatam.

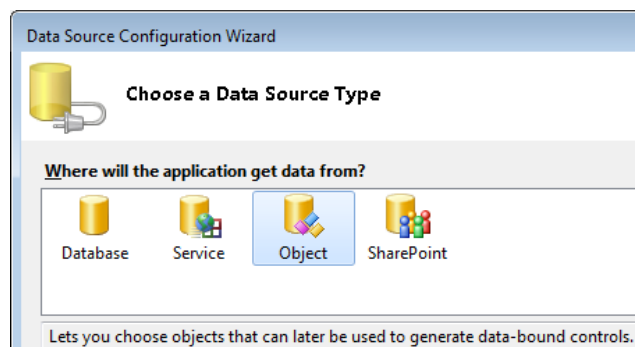
Dati pārskatam tiek ņemti no loga **Data Sources** reģistrētajiem datu avotiem. Šo logu atver ar galvenās izvēlnes komandu **Data | Show Data Sources**:



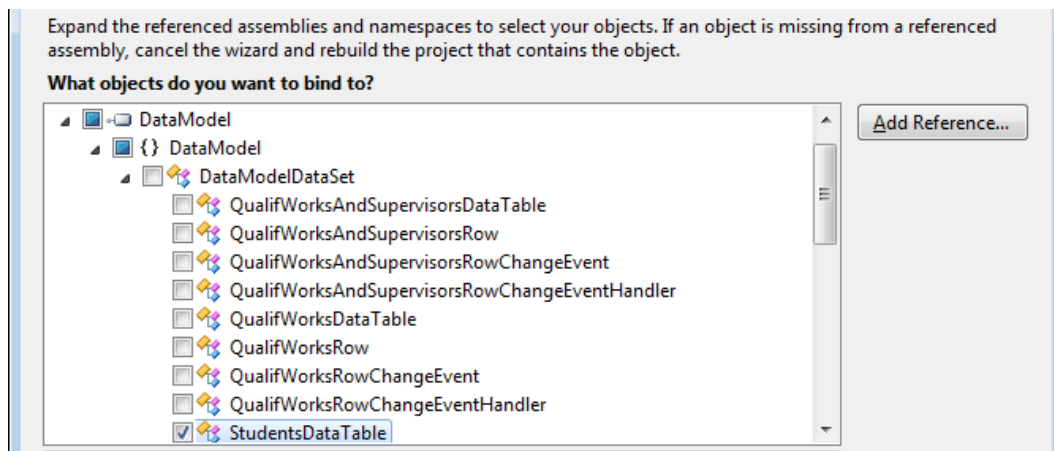
Pēc noklusēšanas šis logs ir tukšs – datu avotu piesaisti veic ar tā rīku joslas komandas **Add New Data Source** palīdzību.

Padoms:** Izstrādes vides versijā Visual Studio 2010, šim nolūkam var izmantot arī iepriekš jau pieminēto logu **Report Data**, izvēloties komandu **New / Dataset...

Piesaistīsim tabulu **Students**, par starpnieku izmantojot projektā **DataModel** definēto tipu – **DataModelDataSet.StudentsDataTable**. Šim nolūkam, nospiežot pogu **Add New Data Source**, atvērt datu avota piesaistes vedni, un tā pirmajā dialogā izvēlēties opciju **Object**:

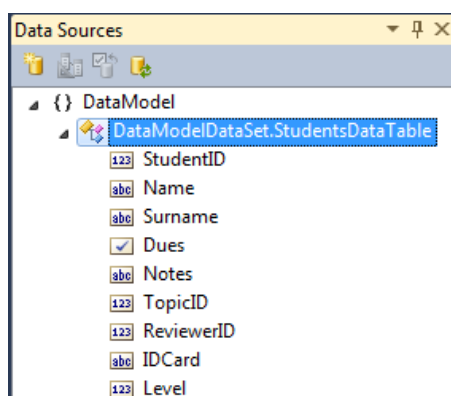


Nospiežot pogu **Next>**, nākošajā dialogā jānorāda tips, kurš nodrošinās piekļuvi datiem, izvēlēties **StudentsDataTable**:

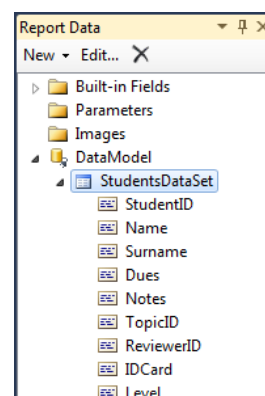


Pēc datu avota objekta izvēles, nospieš pogu **Finish**, pēc kā datu avotu logā būs pieejami izvēlēta tabulas tipa datu lauki:

Visual Studio 2008/2010:



Visual Studio 2010:

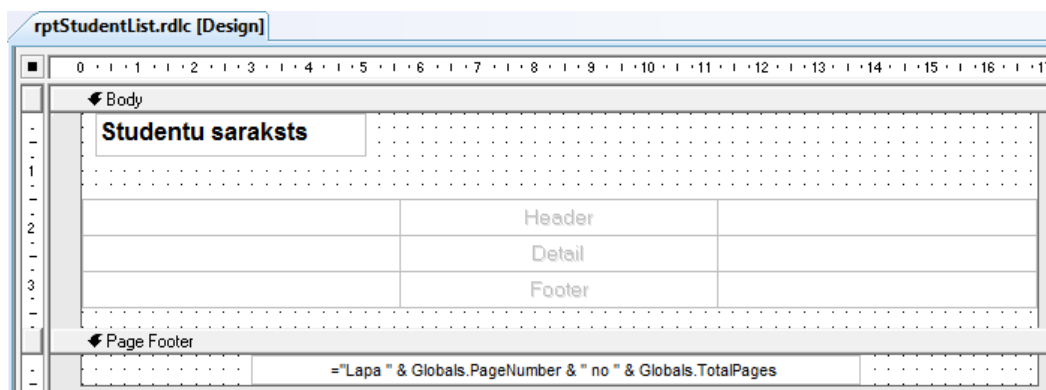


Turpināsim ar pārskata projektēšanu, sekcijā **Body** uzvietojot **Textbox** komponentu, kas atspoguļos pārskata virsrakstu, un uzstādīt īpašības:

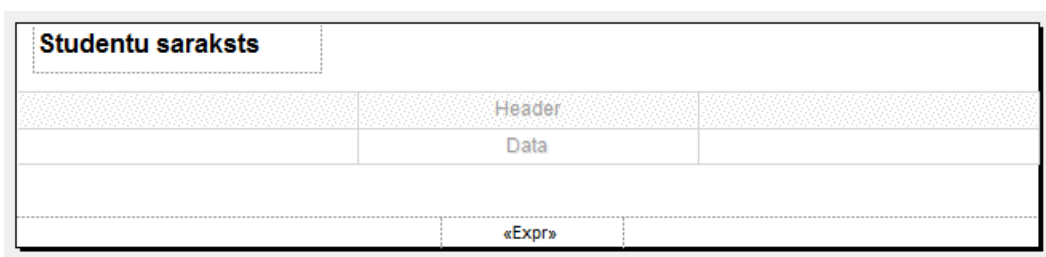
Īpašība:	Vērtība:
Font	Arial; 12pt; Bold
Location	0,25cm; 0cm
Size	4,75cm; 0,75cm
Value	Studentu saraksts

Tabulveida struktūru atspoguļošanai ir paredzēts pārskata komponents **Table** – uzvietot uz pārskata sekcijas **Body** šī komponenta instanci:

- Visual Studio 2008:



- Visual Studio 2010:

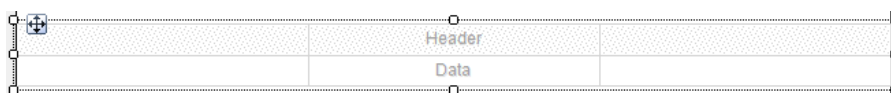


Pēc noklusēšanas projektēšanas posmā komponents **Table** sastāv no galvenes (**Header**), satura (**Detail/Data**) un kājenes (**Footer**) rindām. Galvenes daļā tiek atspoguļota informācija, kas tiks izvadīta tabulas sākumā, kājenes daļā – informācija, kas tiks izvadīta tabulas beigās, bet satura daļā – katrs datu avota raksts tiks izvadīts jaunā šīs sekcijas rindā. Kolonnu platumus un rindas augstumu var mainīt patvaļīgi vai nu ar peles palīdzību, vai arī iezīmējot uz kolonnas/rindas izvēles joslas (pelēkā zona virs un pa kreisi no tabulas šūnām), un īpašību logā norādot atribūta **Width/Height** vērtības.

Iezīmējot noteiktu tabulas kontroles šūnu, tās īpašību logā var norādīt šīs šūnas īpašības. Atribūtā **Value** ir jānorāda izteiksme, kuras vērtība, pārskata sagatavošanas laikā tiks izvadīta šūnā.

Padoms: Izstrādes vides Visual Studio 2010 versijā atribūts **Value** nav tiešā veidā pieejams īpašību logā – ja pārskata komponenta vērtība ir atšķirīga no kāda datu avota lauka, tad šajā komponentā tiek izvadīts teksts **<<Expr>>**, uz kura uzpiežot ar peles dubultklikšķi, atveras lauka vērtības izteiksmes redaktora logs, kurā var redzēt lauka vērtības faktisko izteiksmi, kā arī to modificēt.

Izvēlēties kontroli **table1** un nomainīt tā atribūta **Name** vērtību uz **StudentuSaraksts**. Par faktu, ka ir izvēlēts tabulas komponents nevis kāda tā šūna, liecina tas, ka netiek rādītas pelēkās joslas virs un pa kreisi no tabulas, bet tai apkārt ir bieza, punktota līnija:



Pirmajā pārskata kolonnā izvadīsim rindu numerāciju. Šim nolūkam iezīmē tabulas galvenes rindas pirmās kolonnas šūnu un ierakstīt vērtību **Nr.p.k.**, bet šīs pašas kolonnas satura laukā ierakstīt izteiksmi: **=RowNumber("StudentuSaraksts") & "."**. Funkcija **RowNumber()** atgriež tekošo rindas numuru parametrā norādītajai tabulas kontrolei.

Iezīmēt visu pirmo kolonnu un uzstādīt platuma atribūta **Width** vērtību uz 1,5cm.

Atlikušajās divās kolonnās no datu avota laukus **Name** un **Surname** uzvietot uz tabulas komponenta satura rindas šūnām – tabulas galvenē automātiski tiks ievietoti lauku nosaukumi, bet satura šūnās – vērtības formā **=Fields!<lauks>.Value**, kur **<lauks>** vietā būs atbilstošā tabulas lauka nosaukums. Caur šāda tipa izteiksmi var piekļūt pārskata tekošās rindas lauka vērtībai, un izmantot to citu izteiksmju sastāvā. Pārsaukt kolonnu virsrakstus attiecīgi uz **Vārds** un **Uzvārds**.

Padoms: *Lai vienā tabulas šūnā izvadītu vairāku lauku vērtības, piemēram, gan vārdu, gan uzvārdu, var norādīt šādu izteiksmi: =Fields!Name.Value & " " & Fields!Surname.Value.*

Tā kā visas tabulas komponenta kolonnas ir aizņemtas, tad jāizmanto iespēja pievienot jaunas. Samazināt esošo kolonnu **Vārds** un **Uzvārds** platumu līdz 2,5cm, un, iezīmējot kolonnu **Uzvārds**, atvērt tās konteksta izvēlni, kurā izvēlēties punktu **Insert Column | Right**. Tabulas komponentam tiks pievienota jauna kolonna. Jaunajai kolonnai no datu avotu loga uzvietot lauku **Dues**.

Nomainīt kolonnas **Dues** galvenes šūnas tekstu uz **Parādi?**, bet satura šūnas vērtību uz **=If(Fields!Dues.Value = True, "Jā", "Nē")**. Šāda satura šūnas modificēšana nepieciešama, lai pārskata izdrukā teksta **True/False** vietā tiktu izvadīts teksts **Jā/Nē**. Uzstādīt šīs kolonnas platumu uz 2cm.

Izveidot kolonnu laukam **Piezīmes**. Uzstādīt šīs kolonnas platumu līdz pārskata platuma beigām.

Pārlicināties par tabulas komponenta izvietojumu sekcijas **Body** ietvaros un visu šūnu fonta formatējumu:

Īpašība:	Vērtība:
Font	Arial; 9pt;
Location	0cm; 1cm
Size	16cm; 0,63cm

Ar peli pielāgot sekciju **Body** un **Page Footer** augstumus, lai nebūtu tukši horizontālie laukumi, bet pārskata kopējais platums nepārsniegtu 169 milimetrus.

Saglabāt pārskatu **rptStudentList.rdlc** un aizvērt tā redaktoru.

Projektam **QualifWorksClient** pievienot jaunu formas artefaktu un uzstādīt īpašības:

Īpašība:	Vērtība:
(Name)	frmReport
Size	800; 480
Text	Studentu saraksts

Uz šīs formas uzvietot **Reporting** kategorijas kontroli **ReportViewer** (Visual Studio 2008 versijā – **MicrosoftReportViewer**), un kontroles izvēlnes **ReportViewer Tasks** kombinētā saraksta **Choose Report** izvēlēties iepriekš izveidoto pārskatu (ievērot, ka pēc pārskata norādīšanas formas komponentu laukumā automātiski tiek ievietots komponents **BindingSource**). Neaizverot komponenta izvēlni, izvēlēties tā punktu **Dock in parent container**, lai kontrole automātiski aizņemtu visu formas klienta daļas laukumu.

Ievērot, ka formas **frmReport** notikuma **Load** apstrādātājā automātiski tiek ierakstīta šāda pirmkoda rinda:

```
this.reportViewer1.RefreshReport();
```

Metode **RefreshReport()** aktualizēs datus no pārskatā norādītā datu avota, lai pie formas parādīšanas pārskatā rādītu arī datus.

Pārsaukt komponentu **reportViewer1** par **rpvReport**.

Papildināt formu **frmReport** ar komponentu **DataModelDataSet** un **StudentsTableAdapter** instancēm, nosaucot tās attiecīgi **dsDataModel** un **taStudents**, un komponentu **StudentsDataTableBindingSource** nosaukt par **bsReport**, un uzstādīt tā īpašības:

Īpašība:	Vērtība:
DataSource	dsDataModel
DataMember	Students

Formas **frmReport** definīciju papildināt ar šādu pirmkodu:

```
namespace QualifWorksClient
{
    public partial class frmReport : Form
    {
        public enum Reports { None, StudentList, Order, Assignment };

        private bool bReady;

        public void SetReport(Reports rep)
        {
            this.SuspendLayout();
            bsReport.DataSource = dsDataModel;
            rpvReport.LocalReport.DataSources.Clear();
            bReady = true;

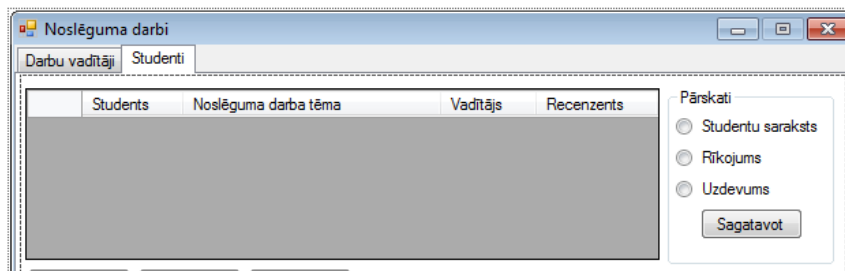
            switch (rep)
            {
                case Reports.StudentList:
                    bsReport.DataMember = "Students";
                    taStudents.Fill(dsDataModel.Students);
                    rpvReport.LocalReport.DataSources.Add(new
                        Microsoft.Reporting.WinForms.ReportDataSource(
                            "DataModel_StudentsDataTable", bsReport));
                    rpvReport.LocalReport.ReportEmbeddedResource =
                        "QualifWorksClient.rptStudentList.rdlc";
                    rpvReport.LocalReport.DisplayName = "Studentu saraksts";
                    this.Text = rpvReport.LocalReport.DisplayName;
                    break;

                default:
                    bReady = false;
                    break;
            }
            this.ResumeLayout(false);
        }

        public frmReport()
        {
            InitializeComponent();
        }

        private void frmReport_Load(object sender, EventArgs e)
        {
            if (bReady)
            {
                this.rpvReport.SetDisplayMode(
                    Microsoft.Reporting.WinForms.DisplayMode.PrintLayout);
                this.rpvReport.RefreshReport();
            }
            else
            {
                throw new Exception(string.Format(
                    "Pārskats '{0}' nav inicializēts!",
                    this.rpvReport.LocalReport.DisplayName));
            }
        }
    }
}
```

Pārskatu izvēli realizēsim formā **frmQualifWorks** ar komponenta **GroupBox** palīdzību, kurā uzvietotas **RadioButton** kontroles un poga kas aktivizē pārskata sagatavošanu un atspoguļošanu uz displeja:



Šim nolūkam, paplašināt formas **frmQualifWorks** platuma īpašību līdz 620 pikseļiem, cilnē **tbpStudents** uzvietot konteineru komponentu **GroupBox** un uzstādīt īpašības:

Īpašība:	Vērtība:
(Name)	grbReports
AutoSize	True
Location	475; 10
Size	122; 130
Text	Pārskati

Konteinerī **grbReports** ievietot trīs **RadioButton** kontroles un uzstādīt īpašības:

Īpašība:	Vērtība:
(Name)	rbStudentList
Location	6; 19
Text	Studentu saraksts

Īpašība:	Vērtība:
(Name)	rbOrder
Location	6; 42
Text	Rīkojums

Īpašība:	Vērtība:
(Name)	rbAssignment
Location	6; 65
Text	Uzdevums

Konteinerī **grbReports** ievietot kontroli **Button** un stādīt šādas īpašības:

Īpašība:	Vērtība:
(Name)	btnPrepareReport
Location	24; 88
Text	Sagatavot

Pogai **btnPrepareReport** izveidot notikuma **Click** apstrādātāju ar šādu realizāciju:

```
private void btnPrepareReport_Click(object sender, EventArgs e)
{
    try
    {
        frmReport rpt = new frmReport();

        if (rbStudentList.Checked)
        {
            rpt.SetReport(frmReport.Reports.StudentList);
        }
        else if (rbOrder.Checked)
        {
            rpt.SetReport(frmReport.Reports.Order);
        }
    }
    catch { }
}
```

```

        {
            throw new Exception("Šis pārskats vēl nav realizēts!");
        }
        else if (rbAssignment.Checked)
        {
            throw new Exception("Šis pārskats vēl nav realizēts!");
        }
        else
        {
            throw new Exception("Norādiet atskaites veidu!");
        }

        rpt.ShowDialog();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

```

Palaist risinājumu uz izpildi un pārlicināties, ka pārskats 'Studentu saraksts' tiek sagatavots un atspoguļots, atbilstoši pārskata dizainam.