

به نام خدا

سند پروژه نهایی درس سیستم های تصمیم یار

دانشجو: محمد تنهایی ۸۷۲۰۲۴۴۶ tanhaei@ce.sharif.edu

استاد: دکتر جعفر حبیبی

دستیار استاد: سهیل حساس یگانه hassas@ce.sharif.edu

هدف پروژه : هدف از انجام این پروژه آشنایی با چگونگی راه اندازی و استفاده از انباره داده جهت تصمیم یاری است.

۱ مستندات نصب :

۱.۱ نصب نرم افزار های مورد نیاز :

در این پروژه سعی داریم از نرم افزار های کد باز استفاده کنیم . برای راه اندازی یک سیستم BI نیاز به وجود یک پایگاه داده ، مفسر جاوا و نیز استفاده از پرتال هایی جهت اجرای نرم افزار اصلی داریم . برای این منظور ما از پایگاه داده های MySQL و مفسر جاوا نسخه ۱.۶ استفاده می کنیم . بستر کاری ما ویندوز است . نیاز به نصب جاوا از آن جهت است که نرم افزار اصلی و همچنین پرتال های ما تحت پلتفرم جاوا ساخته شده اند و بدون جاوا امکان استفاده از آنها وجود ندارد .

۱.۱.۱ نصب MySQL

نصب MySQL با استفاده از ویزارد نصب آن کار بسیار ساده است ، پس از نصب می توانیم با کاربر root و با رمزی که برای آن تعیین کرده ایم به MySQL وارد شویم . با توجه به سادگی کار نصب از توضیح واضحات خودداری می شود ما در این پروژه از MySQL نسخه ۵ استفاده می کنیم.

۱.۱.۲ نصب جاوا

نصب جاوا نیز به مانند کار قبلی کاری ساده است ، تنها نکته با اهمیت اطمینان از وجود JAVA_Path صحیح در سیستم است . در صورتی که مسیر جاوا به درستی تعیین نشده باشد نرم افزار هایی که نیاز به jre دارند نمی توانند به درستی از آن استفاده کنند . برای تعیین java path می توان به local variable های ویندوز مراجعه کرد و متغییر JAVA_PATH را به آن اضافه کرد .

۱.۱.۳ نصب پرتال

برای بخش پرتال می توان از پرتال های tomcat ، jboss و ... استفاده کرد . ما برای این پروژه از پرتال توصیه شده tomcat استفاده می کنیم . برای این منظور نوع خاصی از این پرتال که exo-portal روی آن نصب شده را از اینترنت دریافت نموده ایم . کافی است که این فایل فشرده شده را از حالت فشرده خارج کنیم . برای افزودن کاربر می توان فایل /conf/tomcat-users.xml را به شیوه زیر دست کاری کرد:

```
<tomcat-users>
  <role rolename="admin"/>
  <user username="admin" password="mohammad" roles="admin"/>
</tomcat-users>
```

در اینجا ما یک role به نام admin تعریف کرده ایم و یک کاربر با نام کاربری admin و رمز mohammad و نقش admin تعیین کرده ایم .

پرتال ما آماده اجرا شدن است . می توانیم با اجرای `/bin/exo-run.bat` آنرا اجرا کنیم . تنظیمات اضافی ای که باید روی پرتال انجام دهیم را در بخش نصب نرم افزار اصلی توضیح می دهیم .

۱.۲ نصب SpagoBI

در این قسمت چگونگی نصب SpagoBI را توضیح می دهیم .

بعد از نصب پرتال tomcat نوبت به نصب SpagoBI می رسد . برای نصب کافی است که محتویات فایل `spagobi.zip` را در داخل شاخه نصب پرتال کپی کنیم . حال نوبت به مراحل بعد می رسد .

۱.۲.۱ آماده سازی پایگاه داده

برای این منظور باید فایل `DB.zip` را از حالت فشرده خارج کنیم . حال ۳ فایل `sql` مشاهده می کنیم ، وارد `MySQL` می شویم و با دستور `Source filename.sql` فایل های اسکریپتی را اجرا می کنیم . پس از اجرای این ۳ فایل یک سری جدول و پایگاه داده در `MySQL` ایجاد می شود . عبارتند از : `metadata DB` و پایگاه داده های `Foodmart` و `Bam` که شامل یک سری مثال هستند.

۱.۲.۲ تنظیم ارتباط با پایگاه داده

فایل `EXO-HOME\conf\server.xml` را باز کرده و مشخصات مربوط به `user` و `pass` اتصال به `MySQL` را بررسی می کنیم . `user` پیش فرض برای ارتباط با پایگاه داده ها `root` است . می توان برای این `user` یک رمز تعیین کرد که در حالت پیش فرض بدون رمز در نظر گرفته می شود . نمونه ای از فایل `server.xml` را در زیر مشاهده می کنید:

```
<Resource auth="Container" name="jdbc/spagobi" type="javax.sql.DataSource"/>
<ResourceParams name="jdbc/spagobi">
    ....
    <parameter>
        <name>username</name>
        <value>root</value>
    </parameter>
    <parameter>
        <name>password</name>
        <value> </value>
    </parameter>
    ....
</ResourceParams>
```

۱.۲.۳ شروع کار با SpagoBI

پس از مراحل ساده بالا SpagoBI آماده کار است . می توانید با اجرای فایل `\bin\exo-run.bat` این برنامه را اجرا کنید . برای ورود می توان از رمز های زیر استفاده کرد . (برنامه در آدرس <http://localhost:8080/sbiportal> آماده ورود است) :

<code>biadmin / biadmin</code>	<code>bidev / bidev</code>	<code>biuser / biuser</code>	<code>bimodel / bimodal</code>
<code>bitest / bitest</code>	<code>bidemo / bidemo</code>		

۱.۲.۴ خاتمه کار با SpagoBI

می توان با اجرای `\bin\shutdown.bat` به اجرای برنامه خاتمه داد.

۱.۳ آماده سازی داده ها

برای آماده کردن داده ها برای استفاده در SpagoBI چند راه حل ساده داریم . یکی استفاده از ETL هایی است که کاربرد مخصوص این کار را دارند و می توانند داده ها را از فایل ورودی خوانده ، یک سری فیلتر روی آن انجام دهند و داده ها را داخل پایگاه داده بارگذاری کنند و دیگری ساخت فایل sql از روی فایل داده شده و وارد کردن داده ها است .

با توجه به ساده بودن فایل داده ها روش دوم در زمان سریعتری جواب می دهد ، برای این منظور یک کد ساده نوشتیم که هر خط از ورودی را می خواند و کد لازم را تولید می کرد .

۱.۳.۱ ساخت پایگاه داده

قبل از هر کاری باید یک پایگاه داده برای نگهداری داده ها بسازیم ، ما این پایگاه داده را DSS نام گذاشته ایم .

۱.۳.۲ ساخت جدول های لازم

با توجه به داده های داده شده ۲ جدول `agri` و `person` را ساخته ایم . `agri` شامل فیلد های شهر ، سال ، میزان تولید ، میزان فروش و نوع محصول می باشد . همچنین `person` شامل کد ملی ، جنس ، شهر تولد و سال تولد است .

۱.۳.۳ تمیز کردن داده ها

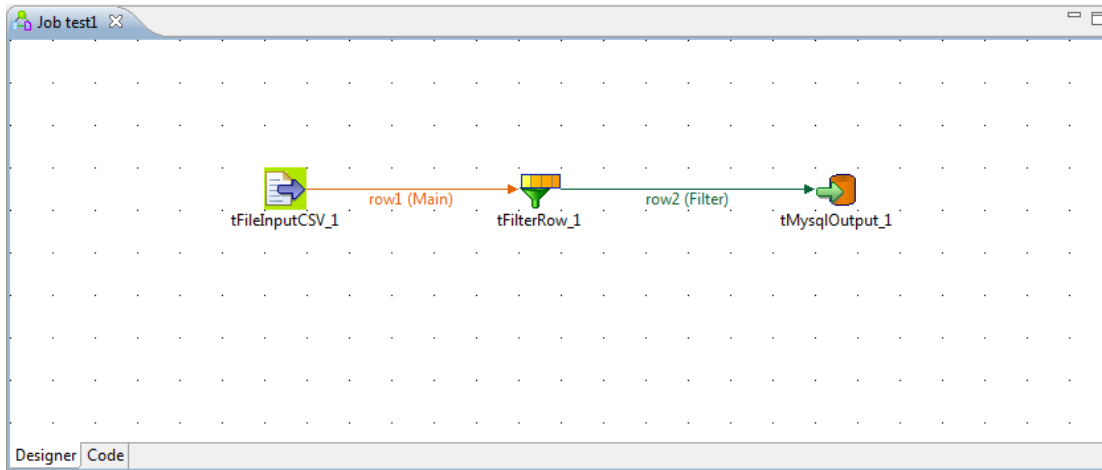
برخی از داده ها دارای کد ملی نبودند و برخی دیگر تاریخ تولد چند رقمی (بزرگتر از ۴ داشتند) ما در این پروژه ردیف هایی که دارای کد ملی نبودند را حذف کرده و از تاریخ تولد های چند رقمی تنها ۴ رقم اول را وارد پایگاه داده ها کرده ایم .

۱.۳.۴ ساخت فایل sql و وارد کردن آن

فایل های `pi.sql` و `agri.sql` را از روی فایل های داده شده و با توجه به تمیز کردن داده ها ساخته ایم . کافی است که با دستور `source` آنها را وارد پایگاه داده های MySQL کنیم .

۱.۳.۵ مختصری از استفاده از ETL

برای استفاده از ETL باید از یک بلوک ورودی ، یک فیلتر و یک بلوک خروجی استفاده کنیم . به این منظور می توانیم در داخل Schema فایل ورودی نوع جدا کننده و نوع داده ها را مشخص کنیم . در بلوک فیلتر نوع فیلتر (مثلاً کد ملی هایی که Null هستند را حذف کنیم) را تعیین می کنیم و در خروجی هم نوع پایگاه داده ها ، اطلاعات اتصال و جدول مقصد را تعیین کرده و دکمه Run را می زنیم . توجه داریم که نوع `encoding` باید برابر با UTF-8 باشد . به همین سادگی می توان از ETL استفاده کرد . تصویر زیر گویای قضایا است!



شکل ۱

۲ مستندات کاربری

در بخش دوم از این سند اندکی در رابطه با کار با ابزار SpagoBI توضیح می دهیم. SpagoBI یک بستر کاری برای انجام پروژه های Business Intelligence است. Spago یک سری ابزار و موتور برای این کارها ارائه می دهد. تصویر زیر گویای مهمترین کارهای Spago است:

Report

Product	Store	Measures		
		Unit Sales	Store Cost	Store Sales
+Canned	+All Stores	841	298.30	1,300.11
+Checkout	+All Stores	1,779	770.12	3,767.71
+Health and Hygiene	+All Stores	10,204	4,603.99	32,571.89
+Bathroom Products	+All Stores	5,885	1,710.32	11,881.60
	+Canada			
	+USA	5,885	1,710.32	11,881.60
	+Mexico			
+Cold Remedies	+All Stores	1,775	490.32	3,350.71
+Decongestants	+All Stores	1,813	412.29	3,300.54
+Hygiene	+All Stores	3,550	645.36	6,002.45
+Pain Relievers	+All Stores	2,054	1,420.58	7,370.58
+Household	+All Stores	27,038	6,993.88	60,460.89
+Personal Care	+All Stores	4,294	1,929.83	9,086.76

OLAP

Data Mining



Dashboard

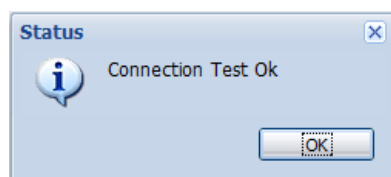
QbE – Query by Example

شکل ۲



در این پروژه ما تنها با ابزار های Report و OLAP کار می کنیم. به صورت خلاصه نحوه ساخت یک Report و یک OLAP را در Spago شرح می دهیم:

۲.۱ افزودن Data source

قبل از توضیح چگونگی ساخت گزارش و تحلیل های OLAP شیوه معرفی یک منبع داده را آموزش می دهیم . برای این منظور می توانیم از منوی Resources گزینه Data Source Configuration را انتخاب کنیم.  را انتخاب کرده و اسم projectDSS را برای این منبع داده ها انتخاب می کنیم . نوع DIALECT را MySQL انتخاب می کنیم ، نوع را With Analytical Drivers و URL را برابر با jdbc:mysql://localhost:3306/project می گیریم ، توجه کنید که project نام پایگاه داده ای است که ما می خواهیم به spago معرفی کنیم . از اسم کاربری root و بدون رمز استفاده می کنیم ، Driver را برابر با org.hsqldb.jdbcDriver قرار می دهیم . حال نوبت به تست  اتصال می رسد اگر همه چیز رو به راه باشد ما شکل زیر (شکل ۳) را مشاهده می کنیم که نشان دهنده موفقیت است .



شکل ۳

حال منبع داده ساخته شده را با دکمه  ذخیره کرده و با فشار دادن  به صفحه قبلی بر می گردیم .


۲.۲ ساخت Report


برای ساخت یک گزارش باید چند مرحله کار انجام دهیم :

۱. ساخت Template برای گزارش
۲. ساخت پارامتر های گزارش
 - a. ساخت لیست مقدار ها
 - b. ساخت محدودیت ها
۳. ثبت کردن یک Analytical Document
 - a. افزودن Template
 - b. دادن پارامتر ها
۴. تست گزارش و اجرای گزارش

SpagoBI دارای دو موتور گزارش است ، یکی از آنها متعلق به Jasper است و دیگری کاری از گروه bird است که تحت بستر eclips اجرا می شود . ما از ابزار های ireport و یا bird برای ساخت template گزارش خود استفاده می کنیم . خروجی این نرم افزار ها یک فایل XML خواهد بود.

در بخش پارامتر ها می توان یک سری پارامتر را به گزارش اضافه کرد ، پارامتر ها در هنگام اجرا مقدار به خود می گیرند و می توانند حالت دینامیکی به گزارش های ما بدهند ، با توجه به اینکه داده های ما به روز نمی شوند نیاز به استفاده از این قابلیت ها نداریم و به مرحله بعدی یعنی ثبت کردن Analytical Document می رویم .

برای ثبت Analytical Document باید از Analytical Models گزینه Documents Configuration را انتخاب کنیم . حال می توانیم با کلیک روی دکمه  یک سند تحلیلی جدید اضافه کنیم . label و name را به دلخواه پر می کنیم و Type را روی Rerpot و engine را روی bird و یا jasper می گذاریم (بسته به نوع xml ای که تولید کرده ایم) . در بخش template فایل xml ای را که ساخته ایم به ابزار معرفی کرده و نوع سند را در بخش سمت راست تعیین کرده و پروژه را ذخیره می کنیم .

مرحله آخر تست گزارش است که در پنجره object list و با آیکن  قابل انجام است .

۲.۳ ساخت یک OLAP

ابتدا باید یک Cube تدارک ببینیم . برای این کار ابزار اتوماتیک خاصی یافت نکردیم و به صورت دستی آن را ساخته ایم بخش هایی از فایل های به صورت زیر است (فایل person.xml)

```
<?xml version="1.0"?>
<Schema name="Person">
  <Dimension name="City">
    <Hierarchy hasAll="true" allMemberName="All Cities" primaryKey="city">
      <Table name="picity"/>
      <Level name="Name" column="city" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Tarikh Tavalod">
    <Hierarchy hasAll="true" allMemberName="All Years" primaryKey="sal">
      <Table name="pitavalod"/>
      <Level name="Name" column="sal" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Sex">
    <Hierarchy hasAll="true" allMemberName="Both" primaryKey="sex">
      <Table name="pisex"/>
      <Level name="Name" column="sex" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Cube name="Tavalod">
    <Table name="person"/>
    <DimensionUsage name="City" source="City" foreignKey="city"/>
    <DimensionUsage name="Tarikh Tavalod" source="Tarikh Tavalod"
foreignKey="tavalod"/>
    <DimensionUsage name="Sex" source="Sex" foreignKey="sex"/>
    <Measure name="Tedad" column="city" aggregator="count" formatString="#,###"/>
  </Cube>
</Schema>
```



برای ساخت چنین فایلی نیاز است چند کار انجام دهیم . ابتدا باید جدول های مربوطه را در پایگاه داده ها ایجاد کنیم . برای نمونه در اینجا نیاز داریم که جدول های pisex ، pitavalod و picity را با یک فیلد که همان کلید اصلی است بسازیم و آنرا با داده ها جدول

person پر کنیم . هر کدام از این جدول ها یک dimension هستند و برای olap از آنها استفاده می کنیم . در Cube ساخته شده باید کلید های خارجی را تعریف کنیم . همچنین باید تعدادی Measure ندارک ببینیم . Measure ها همان ها هستند که می خواهیم از داده ها استخراج کنیم . به عنوان نمونه تعداد تولد. با استفاده از ابزار spago می توان این تعداد را برای هر dimension به دست آورد و اعمال drill down و ... را روی آن انجام داد . این فایل را در /WEB-INF/queries/ قرار می دهیم .

برای ساخت یک تحلیل OLAP نیز مانند ساخت یک گزارش باید یک فایل قالب تهیه کنیم . این فایل را به صورت زیر تهیه کرده ایم

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<olap>
<cube reference='/WEB-INF/queries/person.xml' />
<MDXquery>
select {[Measures].members} ON COLUMNS,
  {[City].[All Cities], [Tarikh Tavalod].[All Years], [Sex].[Both]}} ON ROWS
from [Tavalod]
</MDXquery>
</olap>
```

بخش جدید در این فایل MDXquery است که دارای استاندارد است . در اینجا به معنی انتخاب همه Measures ها در ستون و dimension های City و تاریخ تولد و جنسیت در سطر ها است . توجه کنید که آدرس فایل xml همان است که در مرحله قبلی در /WEB-INF/queries/ قرار دادیم .

مرحله بعد ثبت کردن پارامتر های احتمالی است که در اینجا ما هیچ پارامتری نداریم (با توجه به استاتیک بودن داده ها نیازی به پارامتر نداریم!) در مرحله بعد می توانیم با کلیک روی دکمه  یک سند تحلیلی جدید اضافه کنیم . label و name را به دلخواه پر می کنیم و Type را روی OLAP انتخاب کرده و engine را برابر JPivot قرار می دهیم حال نوبت به انتخاب Data source می رسد . در بخش ابتدایی شیوه افزودن data source را توضیح دادیم ، از data source افزوده شده استفاده می کنیم (به نام projectDSS) و در نهایت فایل قالب که ساخته ایم را در template آپلود می کنیم و دکمه ذخیره را می زنیم . حال می توانیم در پنجره object list و با آیکن  این تحلیل را اجرا کنیم.

۳ سناریوهای در نظر گرفته و فلسفه انتخاب آنها

در این بخش تعدادی سناریو برای تحلیل داده ها در نظر می گیریم ، با توجه به اینکه اطلاعات موالیید سال های بعد از ۷۰ را نداریم نمی توانیم سناریوهایی طراحی کنیم که هر دو جدول داده ها را در نظر بگیرد و به همین خاطر تحلیل های خود را برای هر جدول به صورت جداگانه در نظر می گیریم . سناریوهایی که در نظر می توان گرفت به صورت زیر است :

- برای جدول افراد می توان میزان افزایش جمعیت را بر اساس جنس افراد ، سال و شهر تولد آنها مورد تحلیل قرار داد.
- برای جدول محصولات می توان میزان فروش و تولید را بر اساس شهر ، سال و نوع محصول تحلیل کرد.
- میزان کل فروش و یا تولید را نسبت به سال قبل از آن می توان به صورت یک dashboard نمایش داد .

۳.۱ تحلیل افزایش جمعیت بر اساس شهر ، سال و جنس

برای تحلیل بر اساس شهر ، سال و جنس باید یک تحلیل olap بسازیم . برای این منظور یک فایل xml برای تحلیل olap می سازیم و آنرا به شیوه گفته شده در بخش قبل ثبت می کنیم . این فایل به شکل زیر خواهد بود

```
<?xml version="1.0"?>
<Schema name="Person">

  <Dimension name="City">
    <Hierarchy hasAll="true" allMemberName="All Cities" primaryKey="city">
      <Table name="picity"/>
      <Level name="Name" column="city" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Tarikh Tavalod">
    <Hierarchy hasAll="true" allMemberName="All Years" primaryKey="sal">
      <Table name="pitavalod"/>
      <Level name="Name" column="sal" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Sex">
    <Hierarchy hasAll="true" allMemberName="Both" primaryKey="sex">
      <Table name="pisex"/>
      <Level name="Name" column="sex" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Cube name="Tavalod">
    <Table name="person"/>
    <DimensionUsage name="City" source="City" foreignKey="city"/>
    <DimensionUsage name="Tarikh Tavalod" source="Tarikh Tavalod"
foreignKey="tavalod"/>
    <DimensionUsage name="Sex" source="Sex" foreignKey="sex"/>
    <Measure name="Tedad" column="city" aggregator="count" formatString="#,###"/>
  </Cube>

</Schema>
```

تنها مورد قابل تحلیل با توجه به داده های موجود تعداد موالید در هر سال است . برای این تحلیل باید یک query ترتیب ببینیم . ما در فایل xml بالا یک Measure به نام tedad تعریف کرده ایم که میزان تولد را در هر شهر جمع می کند . فایل query ما به صورت زیر است :

```

select {[Measures].[Tedad]} ON COLUMNS,
Hierarchize(Union(Crossjoin({[City].[All Cities]}, Union(Crossjoin({[Tarikh Tavalod].[All Years]}, {[Sex].[Both]})), Crossjoin({[Tarikh Tavalod].[All Years]}, [Sex].[Both].Children))), Crossjoin([City].[All Cities].Children, Union(Crossjoin({[Tarikh Tavalod].[All Years]}, {[Sex].[Both]}), Crossjoin({[Tarikh Tavalod].[All Years]}, [Sex].[Both].Children)))) ON ROWS
from [Tavalod]

```

فلسفه انتخاب این سناریو هم مقایسه میزان تولد در سال های مختلف ، بین دختر و پسر و با توجه به شهر تولد آن هاست .

نمونه ای خروجی تحلیلی ما به صورت زیر است(شکل ۴) :

			Measures
City	Tarikh Tavalod	Sex	Tedad
-All Cities	+All Years	-Both	100,103
		false	49,905
		true	50,198
آبادان	+All Years	+Both	11,111
اصفهان	+All Years	+Both	11,115
تهران	+All Years	+Both	11,104
سیزوار	+All Years	+Both	11,255
شیراز	+All Years	+Both	11,057
مشهد	+All Years	+Both	11,221
تیریز	+All Years	+Both	11,043
کرج	+All Years	+Both	11,190
تندیس	+All Years	+Both	11,007

شکل ۴

۳.۲ تحلیل میزان تولید در سال ها مختلف بر اساس شهر و نوع محصول

این تحلیل نیز از نوع Olap است ، در تحلیل های OLAP می توان اعمالی مانند drill down و ... انجام داد ، در این تحلیل هدف ما فراهم آوردن امکان مقایسه میزان تولید و فروش در سال های مختلف با قابلیت ریز شدن بر روی شهر تولید کننده و نوع محصول است . با توجه به ساده بودن جدول داده ها تنها می توان سناریو های محدودی مانند مجموع تولید ، فروش و میزان سود که از تقسیم میزان فروش به تولید به دست می آید .

فلسفه انتخاب این نوع تحلیل مقایسه میزان تولید و فروش در سال های مختلف ، ضرر و یا نفع در سال های مختلف و بر اساس شهر ها و محصولات متفاوت است .

```

<?xml version="1.0"?>
<Schema name="AGRI">

  <Dimension name="City">
    <Hierarchy hasAll="true" allMemberName="All Cities" primaryKey="city">
      <Table name="cities"/>
      <Level name="Name" column="city" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Date">
    <Hierarchy hasAll="true" allMemberName="All Years" primaryKey="sal">
      <Table name="productdates"/>
      <Level name="Name" column="sal" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Dimension name="Product">
    <Hierarchy hasAll="true" allMemberName="All Products" primaryKey="kind">
      <Table name="productkinds"/>
      <Level name="Name" column="kind" uniqueMembers="false"/>
    </Hierarchy>
  </Dimension>

  <Cube name="Foroosh">
    <Table name="agri"/>
    <DimensionUsage name="City" source="City" foreignKey="city"/>
    <DimensionUsage name="Product" source="Product" foreignKey="no"/>
    <DimensionUsage name="Date" source="Date" foreignKey="sal"/>
    <Measure name="Foroosh" column="foroosh" aggregator="sum"
formatString="#,###"/>

    <Measure name="Tolid" column="tolid" aggregator="sum" formatString="#,###"/>

    <CalculatedMember name="sood" dimension="Measures">
      <Formula>
        ([Measures].[Foroosh] / [Measures].[Tolid] -1)
      </Formula>
      <CalculatedMemberProperty name="FORMAT_STRING" value="%#,##0.00"/>
    </CalculatedMember>
  </Cube>

</Schema>

```

ما در این فایل ۳ نوع measure انتخاب کرده ایم ، میزان تولید ، میزان فروش و سود حاصل از فروش .

Query مورد نیاز برای این تحلیل نیز به صورت زیر خواهد بود :

```
select {[Measures].[Foreosh], [Measures].[Tolid], [Measures].[sood]} ON COLUMNS,
  Hierarchize(Union(Crossjoin({[City].[All Cities]}, Union(Crossjoin({[Date].[All Years]}, {[Product].[All Products]})), Crossjoin({[Date].[All Years]}, [Product].[All Products].Children))), Union(Crossjoin([City].[All Cities].Children, Union(Crossjoin({[Date].[All Years]}, {[Product].[All Products]})), Crossjoin({[Date].[All Years]}, [Product].[All Products].Children))), Crossjoin([City].[All Cities].Children, Union(Crossjoin([Date].[All Years].Children, {[Product].[All Products]}), Crossjoin([Date].[All Years].Children, [Product].[All Products].Children)))))) ON ROWS
from [Foreosh]
```

در این query ما میزان فروش ، تولید و سود را در شهر های مختلف و برای محصولات مختلف مقایسه کرده ایم .

نمونه ای از خروجی تحلیل به صورت زیر است (شکل ۵) :

			Measures		
City	Date	Product	Foreosh	Tolid	sood
-All Cities	+All Years	-All Products	518,288,294,089	501,004,326,427	%3.45
		جو	175,234,495,609	166,522,261,127	%5.23
		ذرت	173,635,480,157	166,772,894,668	%4.11
		گندم	169,418,318,323	167,709,170,632	%1.02
ایرادان	-All Years	-All Products	52,770,835,672	56,084,131,229	%- 5.91
		جو	17,380,754,315	21,728,009,276	%- 20.01
		ذرت	17,966,997,114	18,831,331,646	%- 4.59
		گندم	17,423,084,243	15,524,790,307	%12.23
	1360	-All Products	2,462,155,384	1,800,887,438	%36.72
		جو	262,833,223	169,933,851	%54.67
		ذرت	1.005.613.389	1.248.040.165	%- 19.42

شکل ۵

۳.۳ تحلیل میزان کل تولیدات نسبت به سال قبل

این تحلیل ، تحلیلی ساده اما کاربردی است ، و می تواند میزان رشد سال آخر را نسبت به میانگین تولید در سال های قبل نمایش دهد . برای نمایش این تحلیل از یک داشبورد مدیریتی ساده استفاده می کنیم .

این تحلیل هر چند ساده است اما می تواند معیار خوبی برای مقایسه میزان تولید امسال به سال های دیگر باشد و رشد و یا افت تولید را نمایش دهد .

Query مورد نیاز برای به دست آوردن این نسبت به صورت زیر است که آنرا در یک dataset قرار می دهیم :

```

select 100*sum(foroosh)/(
select avg(temp) from(
select sum(foroosh) as temp from agri
where sal != (select max(sal) from agri)
group by sal ) as temp2
)
as value from agri
where sal = (select max(sal) from agri);

```

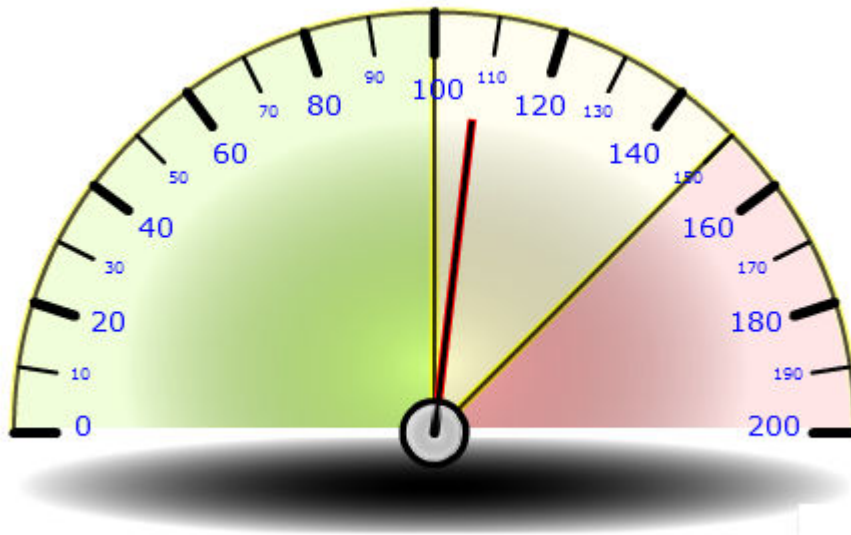
برای ساخت داشبورد قالب زیر را طراحی کرده و در قسمت **template** آدرس آنرا می دهیم :

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<DASHBOARD movie="dashboards/rot.lzx.swf" displayTitleBar="true">
<DIMENSION width="440" height="350" />
<CONF>
<PARAMETER name="minValue" value="0" />
<PARAMETER name="maxValue" value="200" />
<PARAMETER name="lowValue" value="100" />
<PARAMETER name="highValue" value="150" />
</CONF>
<DATA url="/servlet/AdapterHTTP?ACTION_NAME=GET_DATASET_RESULT">
</DATA>
</DASHBOARD>

```

خروجی به صورت زیر خواهد بود (شکل ۶) : که نشان دهنده رشد فروش در سال آخر است.

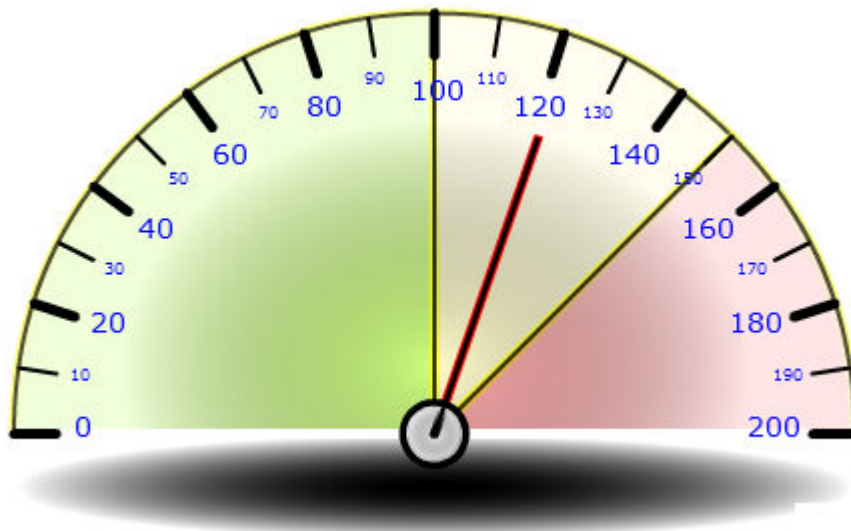


شکل ۶

می توان کاری مشابه برای تولید و نسبت این ۲ انجام داد به عنوان نمونه برای بررسی میزان رشد فروش به تولید امسال نسبت به میانگین فروش به تولید طی سال های گذشته از query زیر استفاده می کنیم :

```
select (
select 100*sum(foroosh)/(
select avg(temp2) from
(
select sum(foroosh) as temp2 from agri
where sal != (select max(sal) from agri)
group by sal ) as temp3 )
as value2 from agri
where sal = (select max(sal) from agri)
) / (
select 100*sum(tolid)/(
select avg(temp) from
(
select sum(tolid) as temp from agri
where sal != (select max(sal) from agri)
group by sal ) as temp2
)
as value1 from agri
where sal = (select max(sal) from agri)
) as value;
```

و خروجی به صورت زیر خواهیم داشت (شکل ۷) :



شکل ۷

۳.۴ گزارش های مختلف

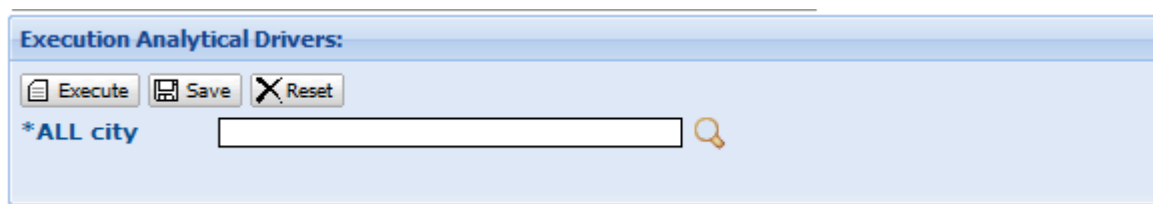
یکی دیگر از انواع تحلیل ها که خیلی حالت داینامیک ندارد گزارش هایی است که می توان برای کاربر صادر کرد .

در اینجا ما یک گزارش ساده داریم که نام شهر را از کاربر گرفته و مشخصات تولید و فروش آن را در سال های مختلف بر اساس نوع محصول نمایش می دهد(شکل ۸) .


agri_foroosh	agri_tolid
agri_sal 1360	
agri_no جو	
7376646875	4872909436
agri_no ذرت	
5781550718	5557942914
agri_no گندم	
7984458830	5336724850
agri_sal 1361	
agri_no جو	

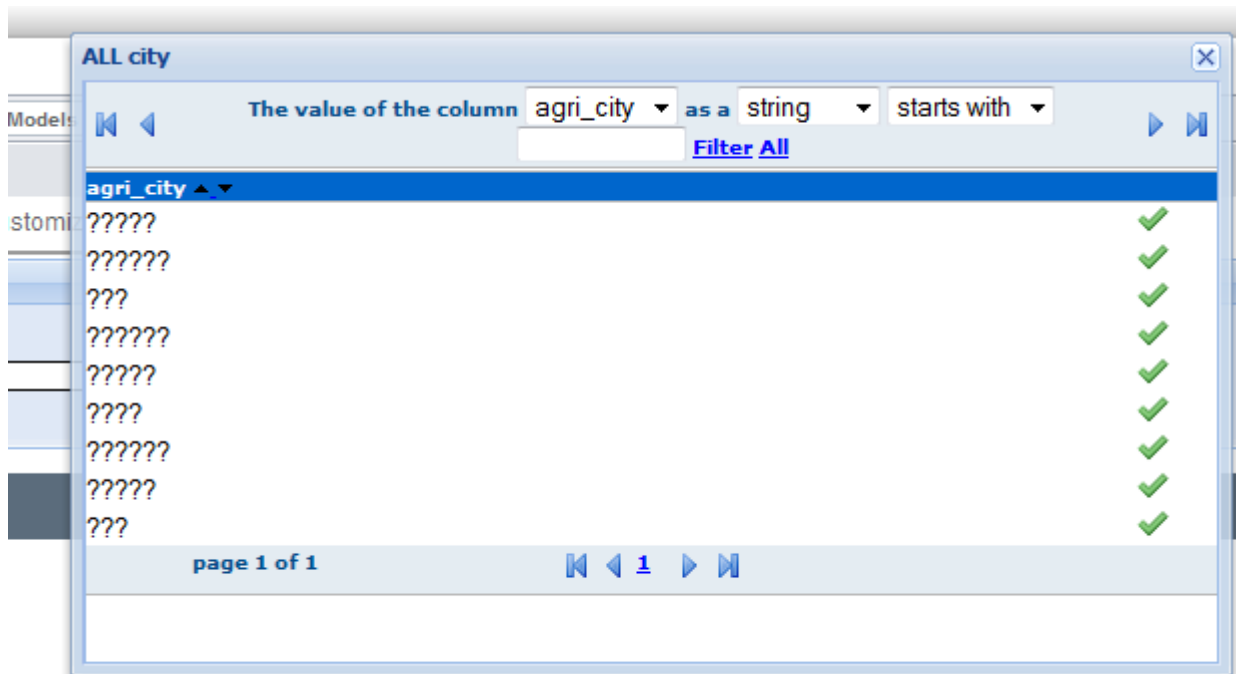
شکل ۸

برای این منظور باید پارامتر تعریف کنیم . پس از تعریف پارامتر و LOV برای گزارش گیری صفحه ای به این صورت باز می شود :



شکل ۹

با انتخاب دکمه  می توان شهر مورد نظر را از لیستی که ساخته ایم انتخاب کرد ، اینجاست که مشکل فارسی نویسی خود را نشان می دهد ما برای این منظور راه حلی ارائه کرده ایم که در تحویل حضوری ارائه خواهد شد و از دراز کردن این سند پرهیز می کنیم .



شکل ۱۰

بعد از تعریف این گزارش که کلیات آن ارائه شد این سند را می بندیم!

ما برای ساخت این سند از ابزار ساخت سند Jasper استفاده کرده ایم . و به خاطر طولانی نشدن مبحث از بحث جدا در مورد آن خودداری می کنیم .

و من الله توفیق

محمد تنهایی اسفند ماه ۱۳۸۷