

第五组Dracula 课程实验总结报告

小组成员：71119104包亦成 71119138王骏

实验任务1_1

1.任务描述

以奥运会为主题确定主题案例信息，确定元素、属性、实体等数据，以及数据间的结构关系，建立初始信息模型，编写XML文档，确保文档是“良好的”。

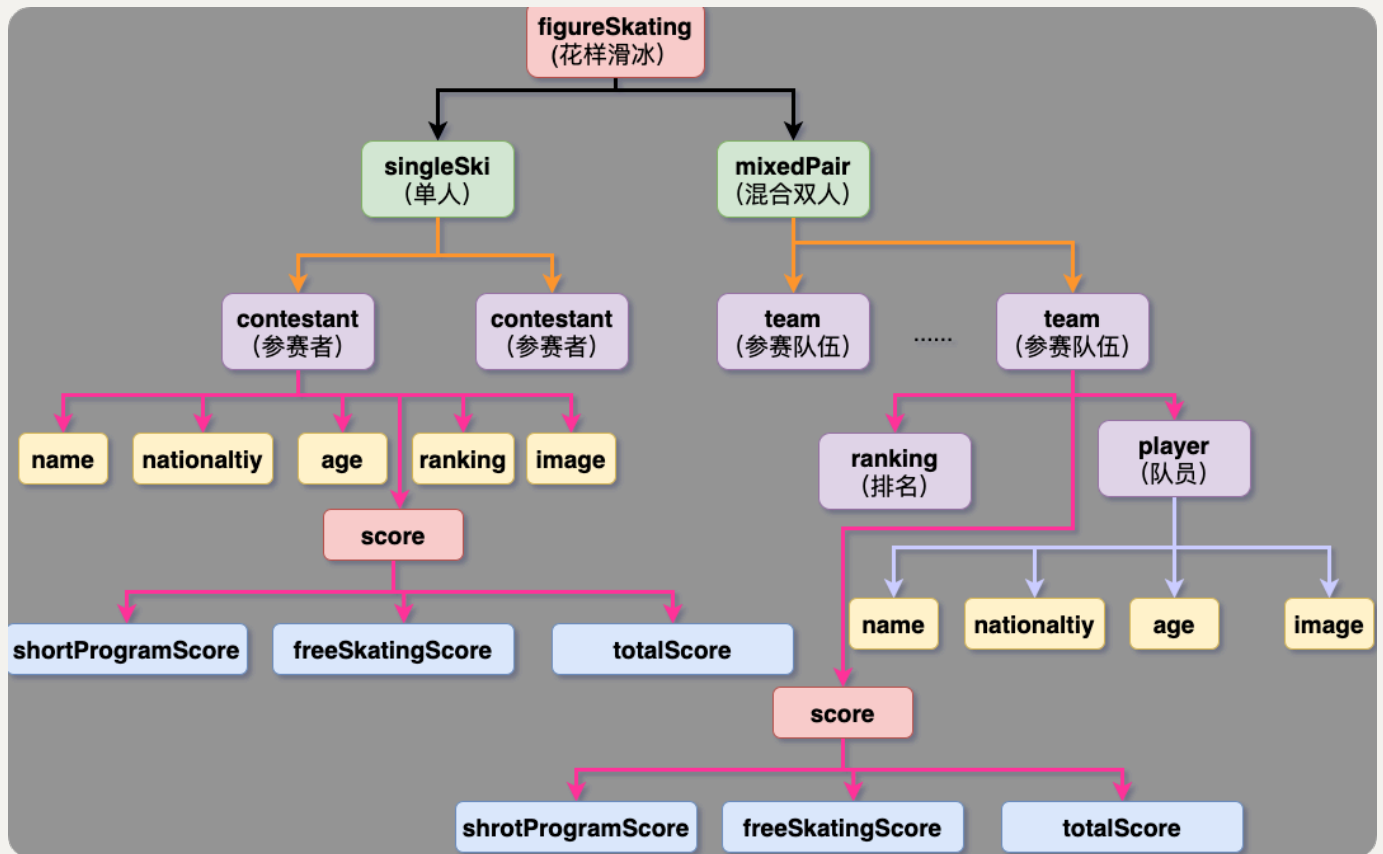
2.主要实验代码(仅展示了部分数据)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE figureSkating SYSTEM "figureSkating.dtd">
<figureSkating year="2022" location="Beijing">
  <single number="2">
    <contestant>
      <name>Boyang JIN</name>
      <nationality>China</nationality>
      <age>24</age>
      <ranking>9</ranking>
      <image>https://olympics.com/beijing-2022/olympic-
games/static/owg2022/img/bios/photos/1058056.jpg</image>
    </contestant>
    <contestant>
      <name>Yuzuru HANYU</name>
      <nationality>Japan</nationality>
      <age>27</age>
      <ranking>4</ranking>
      <image>https://olympics.com/beijing-2022/olympic-
games/static/owg2022/img/bios/photos/1038834.jpg</image>
    </contestant>
  </single>
```

```
<mixedPair teamNum="1">
  <team>
    <ranking>1</ranking>
    <womanPlayer>
      <name>Wenjing SUI</name>
      <nationality>China</nationality>
      <age>26</age>
      <image>https://olympics.com/beijing-2022/olympic-
games/static/owg2022/img/bios/photos/1058064.jpg</image>
    </womanPlayer>
    <manPlayer>
      <name>Cong HAN</name>
      <nationality>China</nationality>
      <age>29</age>
      <image>https://olympics.com/beijing-2022/olympic-
games/static/owg2022/img/bios/photos/1058065.jpg</image>
    </manPlayer>
  </team>
</mixedPair>
</figureSkating>
```

3.效果截图

- 信息模型



实验任务1_2

1.任务描述

针对案例信息模型和XML文档，编写校验所需的DTD文档，确保XML文档是“有效的”

2.主要实验代码

```

<!ELEMENT figureSkating (singleSki,mixedPair)>
<!ATTLIST figureSkating year CDATA #REQUIRED>
<!ATTLIST figureSkating location CDATA #REQUIRED>

<!ELEMENT singleSki (contestant+)>
<!ATTLIST singleSki number CDATA #REQUIRED>

<!ELEMENT contestant (name, nationality, age, ranking, image, score)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT nationality (#PCDATA)>
<!ELEMENT age (#PCDATA)>
<!ELEMENT ranking (#PCDATA)>

```

```

<!ELEMENT image (#PCDATA)>
<!ELEMENT score (shortProgramScore, freeSkatingScore, totalScore)>
<!ELEMENT shortProgramScore (#PCDATA)>
<!ELEMENT freeSkatingScore (#PCDATA)>
<!ELEMENT totalScore (#PCDATA)>

<!ELEMENT mixedPair (team+)>
<!ATTLIST mixedPair teamNum CDATA #REQUIRED>
<!ELEMENT team (ranking, scores, womanPlayer, manPlayer)>
<!ELEMENT womanPlayer (name, nationality, age, image)>
<!ELEMENT manPlayer (name, nationality, age, image)>

```

3.效果截图

DTD可以正确反应文档的错误

```

5  <?xml version="1.0" encoding="UTF-8"?>
4  <!DOCTYPE figureSkating SYSTEM "figureSkating.dtd">
3  <figureSkating year="2022" location="Beijing">
2  <single number="10">
1  <contestant>
  <name>Boyang JIN</name>
  <nationality>China</nationality>
  <age>24</age>
  <ranking>9</ranking>
  <image>https://olympics.com/beijing-2022/olympic-games/static/owg2022
  </contestant>
  <contestant>
  <name>Yuzuru HANYU</name>
  <nationality>Japan</nationality>

```

Dracula1_2.xml 2 X Dracula1_3.xml figureSkating.dtd

week2 >

8 <

7 < View Problem No quick fixes available

6 <figureSkating year="2022" location="Beijing">

5 <single number="10">

4 <contestant>

3 <name>Boyang JIN</name>

实验任务1_3

1.任务描述

针对案例XML文档中的数据，编写所需的Schema文档，确保XML文档是“有效的”

2.主要实验代码

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.w3school.com.cn"
xmlns="http://www.w3school.com.cn" elementFormDefault="qualified">

    <xs:element name="figureSkating">
        <xs:complexType mixed="true">
            <xs:sequence>
                <xs:element name="singleSki" type="singleCompetition"
minOccurs="1" maxOccurs="2" />
                <xs:element name="mixedPair" type="mixedCompetition"
minOccurs="1" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:simpleType name="scoreList">
        <xs:list itemType="xs:integer"/>
    </xs:simpleType>

    <xs:simpleType name="sex">
        <xs:restriction base="xs:string">
            <xs:enumeration value="male" />
            <xs:enumeration value="female" />
        </xs:restriction>
    </xs:simpleType>

    <xs:complexType name="singleCompetition" mixed="true">
        <xs:sequence>
            <xs:element name="contestant" minOccurs="0"
maxOccurs="unbounded" />
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

```

        </xs:sequence>
        <xs:attribute name="number" type="xs:integer" />
        <xs:attribute name="sex" type="sex" />
    </xs:complexType>

    <xs:complexType name="mixedCompetition" mixed="true">
        <xs:sequence>
            <xs:element name="team" minOccurs="1"
maxOccurs="unbounded" type="team" />
        </xs:sequence>
        <xs:attribute name="teamNum" type="xs:integer" />
    </xs:complexType>

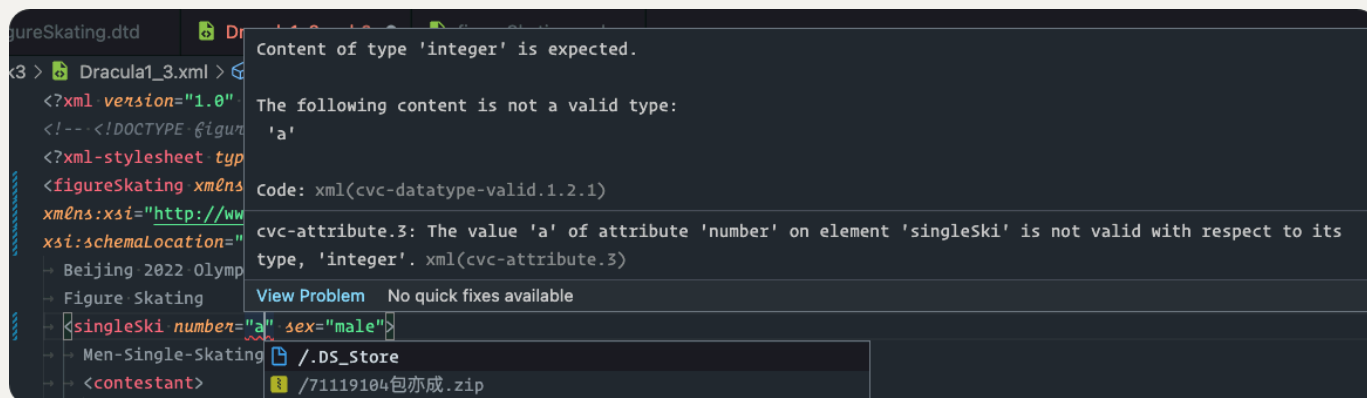
    <xs:complexType name="contestant" mixed="true">
        <xs:all>
            <xs:element name="name" type="xs:string" />
            <xs:element name="nationality" type="xs:string" />
            <xs:element name="age" />
            <xs:element name="ranking" nillable="true" />
            <xs:element name="image" type="xs:string"/>
        </xs:all>
    </xs:complexType>

    <xs:complexType name="team" mixed="true">
        <xs:sequence>
            <xs:element name="ranking" />
            <xs:element name="scores" type="scoreList"/>
            <xs:element name="womanPlayer" type="contestant" />
            <xs:element name="manPlayer" type="contestant" />
        </xs:sequence>
    </xs:complexType>
</xs:schema>

```

3.效果截图

Schema文档可以校验出XML文档中的错误



实验任务2_0

1.任务描述

建立一个信息发布网页

- 安装和配置IIS
- 根据主题，选择XML数据中需要展示的内容
- 利用CSS技术，编写CSS文档，设计显示效果通过不断改进，达到最终理想的结果
- 通过IIS，访问最终的展示效果

2.主要实验代码(最终的CSS代码)

```
figureSkating {  
    position: fixed;  
    left: 10%;  
    right: 10%;  
    padding: 3.5%;  
    font-family: Georgia, 'Times New Roman', Times, serif;  
    position: absolute;  
    display: flex;  
    flex-direction: column;  
    text-align: center;  
    font-size: 45px;  
    box-shadow: 0px 0px 40px #888888;  
    background-image: url(./back4.svg);  
}
```

```
mixedPair,
singleSki {
    margin: 20px;
    font-size: 30px;
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
    box-shadow: 0px 0px 15px #ffffff;
    border-radius: 100px;
    background-color: #71afe558;
}
contestant,
team {
    height: 80px;
    font-size: 20px;
}

name,
nationality,
age,
ranking,
shortProgramScore,
freeSkatingScore,
totalScore {
    color: #008272;
    font-size: 30px;
}
image {
    display: none;
}
```

3.效果截图

通过IIS访问的页面效果

Beijing 2022 Olympic Winter Games Figure Skating

Men-Single-Skating

Name: **Boyang JIN** Country: **China** Age: **24** Ranking: **9** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Yuzuru HANYU** Country: **Japan** Age: **27** Ranking: **4** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Nathan CHEN** Country: **USA** Age: **22** Ranking: **9** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Yuma KAGIYAMA** Country: **Japan** Age: **18** Ranking: **2** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Morisi KVITELASHVILI** Country: **Georgia** Age: **26** Ranking: **10** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Women-Single-Skating

Name: **Anna SHCHERBAKOVA** Country: **Russia** Age: **17** Ranking: **1** 90.98 179.45 270.43

Name: **Kaori SAKAMOTO** Country: **Japan** Age: **21** Ranking: **3** 90.98 179.45 270.43

Name: **Kamila VALIEVA** Country: **Russia** Age: **15** Ranking: **4** 90.98 179.45 270.43

Name: **Alysa LIU** Country: **USA** Age: **16** Ranking: **7** 90.98 179.45 270.43

Name: **Loena HENDRICKX** Country: **Belgium** Age: **22** Ranking: **8** 90.98 179.45 270.43

Pair-Skating

Ranking: **1** 9 8 9 Name: **Wenjing SUI** Country: **China** Age: **26** Name: **Cong HAN** Country: **China** Age: **29**

Ranking: **2** 9 8 7 Name: **Evgenia TARASOVA** Country: **ROC** Age: **27** Name: **Vladimir MOROZOV** Country: **ROC** Age: **29**

Beijing 2022 Olympic Winter Games Figure Skating

Men-Single-Skating

Name: **Boyang JIN** Country: **China** Age: **24** Ranking: **9** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Yuzuru HANYU** Country: **Japan** Age: **27** Ranking: **4** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Nathan CHEN** Country: **USA** Age: **22** Ranking: **9** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Yuma KAGIYAMA** Country: **Japan** Age: **18** Ranking: **2** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Name: **Morisi KVITELASHVILI** Country: **Georgia** Age: **26** Ranking: **10** shortProgramScore: **90.98** freeSkatingScore: **179.45** totalScore: **270.43**

Women-Single-Skating

Name: **Anna SHCHERBAKOVA** Country: **Russia** Age: **17** Ranking: **1** 90.98 179.45 270.43

Name: **Kaori SAKAMOTO** Country: **Japan** Age: **21** Ranking: **3** 90.98 179.45 270.43

Name: **Kamila VALIEVA** Country: **Russia** Age: **15** Ranking: **4** 90.98 179.45 270.43

Name: **Alysa LIU** Country: **USA** Age: **16** Ranking: **7** 90.98 179.45 270.43

实验任务2_1

1.任务描述

针对信息模型和所有XML数据（元素，属性，实体）

使用XSL的XSLT+Xpath技术

*1. 发布小组主页，包括：

小组的名称、logo和简介

小组成员的姓名、学号、个人信息、个人介绍、图片/logo

建立“数据网页”和“宣传网页”的链接

*2. 发布数据网页，包括：

小组所有数据记录，按列表显示

每条数据记录的内容

建立返回主页的链接

*3. 发布宣传网页，包括：

每人选择和设计一条数据记录，制作广告宣传页

建立返回主页的链接

2.主要实验代码

- 主页的XSLT

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
    <body style="display:flex; align-items; center; flex-
direction:row; height:1000px; margin:0">
      <!-- bg -->
      
      <!-- logo -->
      <div style="display:flex; position:relative; margin-
left:100px">
        <p style="position:absolute; top:300px;left:70px;
width:300px">
```

```

        <a href="dataPage.xml"></a>
        <a href="advertisePage.xml" style="margin-left:100px"></a>
    </p>
    
</div>
<!-- personal logo -->
<div style="display:flex; align-items:center; justify-
content:space-around; flex-direction:column; margin-left:50px">
    
    
</div >
<!-- info -->
<xsl:apply-templates select="team/teamMember"/>
</body>
</html>
</xsl:template>

<xsl:template match="team/teamMember">
    <div style="display:flex;flex-direction:column; font-
family:cursive; font-size:35px; margin-top:100px; margin-left:50px">
        <xsl:for-each select="member">
            <div style="margin-bottom:300px">
                <div><xsl:value-of select="name" /></div>
                <div><xsl:value-of select="number" /></div>
                <div><xsl:value-of select="info" /></div>
            </div>
        </xsl:for-each>
    </div>
</xsl:template>

</xsl:stylesheet>

```

- 数据网页XSLT

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:template match="/">
    <html>
      <body style="margin:0; font-size:150%">
        <!-- bg -->
        
        <div style="display:flex; flex-
direction:column;align-items:center; justify-content:center">
          <a href="teamPage.xml" style="margin-top:3vh">
            
          </a>
          <div style="margin-top:3vh; ">
            <table border="1" style="font-size:150%">
              <xsl:copy-of select="$singleHeader" />
              <xsl:apply-templates
select="figureSkating/singleSkiMan" />
            </table>
          </div>

          <div style="margin-top:3vh; ">
            <table border="1" style="font-size:150%">
              <xsl:copy-of select="$singleHeader" />
              <xsl:apply-templates
select="figureSkating/singleSkiWomen" />
            </table>
          </div>

          <div style="margin-top:3vh; ">
            <table border="1" style="font-size:150%">
              <xsl:copy-of select="$mixedHeader" />
              <xsl:apply-templates
select="figureSkating/mixedPair" />
            </table>
          </div>

```

```

        </div>
    </body>
</html>
</xsl:template>

<!-- 男女子单人表 -->
<xsl:template name="singleSki">
    <xsl:if test="./@sex='male'">
        <h2>male</h2>
    </xsl:if>
    <xsl:if test="./@sex='female'">
        <h2>female</h2>
    </xsl:if>
    <xsl:for-each select="contestant">
        <tr>
            <td style="">
                <xsl:value-of select="name" />
            </td>
            <td style="">
                <xsl:value-of select="nationality" />
            </td>
            <td style="">
                <xsl:value-of select="age" />
            </td>

            <xsl:choose>
                <xsl:when test="ranking = 1">
                    <td bgcolor="#f3a694">
                        <xsl:value-of select="ranking" />
                    </td>
                </xsl:when>
                <xsl:otherwise>
                    <td style="">
                        <xsl:value-of select="ranking" />
                    </td>
                </xsl:otherwise>
            </xsl:choose>
        </tr>
    </xsl:for-each>

```

```

</xsl:template>

<!-- 混合双人表 -->
<xsl:template name="mixedPair">
    <h2>Mixed Pair</h2>
    <xsl:for-each select="team">
        <xsl:sort select="ranking"/>
        <tr>
            <xsl:if test="position() mod 2 = 1">
                <xsl:attribute
name="bgcolor">#c7e0f4</xsl:attribute>
            </xsl:if>
            <td style="">
                <xsl:value-of select="ranking" />
            </td>
            <td style="">
                <xsl:value-of select="*/nationality" />
            </td>
            <td style="">
                <div style="display:flex-inline">
                    <div style="margin-right:2px">
                        <xsl:value-of select="womanPlayer/name"/>
                    </div>
                    <div>
                        <xsl:value-of select="manPlayer/name" />
                    </div>
                </div>
            </td>
            <td style="">
                <xsl:value-of select="scores" />
            </td>
        </tr>
    </xsl:for-each>
</xsl:template>

<xsl:variable name="singleHeader">
    <tr bgcolor="#fff">
        <th>name</th>
        <th>country</th>

```

```

        <th>age</th>
        <th>ranking</th>
    </tr>
</xsl:variable>

<xsl:variable name="mixedHeader">
    <tr bgcolor="#fff">
        <th>ranking</th>
        <th>country</th>
        <th>members</th>
        <th>scores</th>
    </tr>
</xsl:variable>

<xsl:template match="figureSkating/singleSkiMan">
    <xsl:call-template name="singleSki">
    </xsl:call-template>
</xsl:template>

<xsl:template match="figureSkating/singleSkiWomen">
    <xsl:call-template name="singleSki">
    </xsl:call-template>
</xsl:template>

<xsl:template match="figureSkating/mixedPair">
    <xsl:call-template name="mixedPair">
    </xsl:call-template>
</xsl:template>

</xsl:stylesheet>

```

- 宣传网页XSLT

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

    <xsl:template match="/">

```

```

<html>
  <body style="margin:0;">
    <!-- bg -->
    
    <!-- logo -->
    
    <!-- info -->
    <div style="padding-top: 5%">
      <!-- home link -->
      <p style=" text-align:center">
        <a href="teamPage.xml">
          
        </a>
      </p>
      <div style="display:flex; flex-direction:column; align-
items:center; justify-content:flex-start">
        <xsl:for-each select="advertises/advertise">
          <div style="text-align:center; font-family:cursive;
font-size:35px;margin-top:40px">
            
            <xsl:value-of select="email" />
          </div>
          <div style="text-align:center; font-family:cursive;
font-size:35px">
            
            <xsl:value-of select="github"/>
          </div>
        </xsl:for-each>
      </div>
    </div>

    <!-- wjinfo -->
    <!-- <div style="padding-top: 5%">
      <div style="text-align:center; font-family:cursive; font-
size:35px;">

```



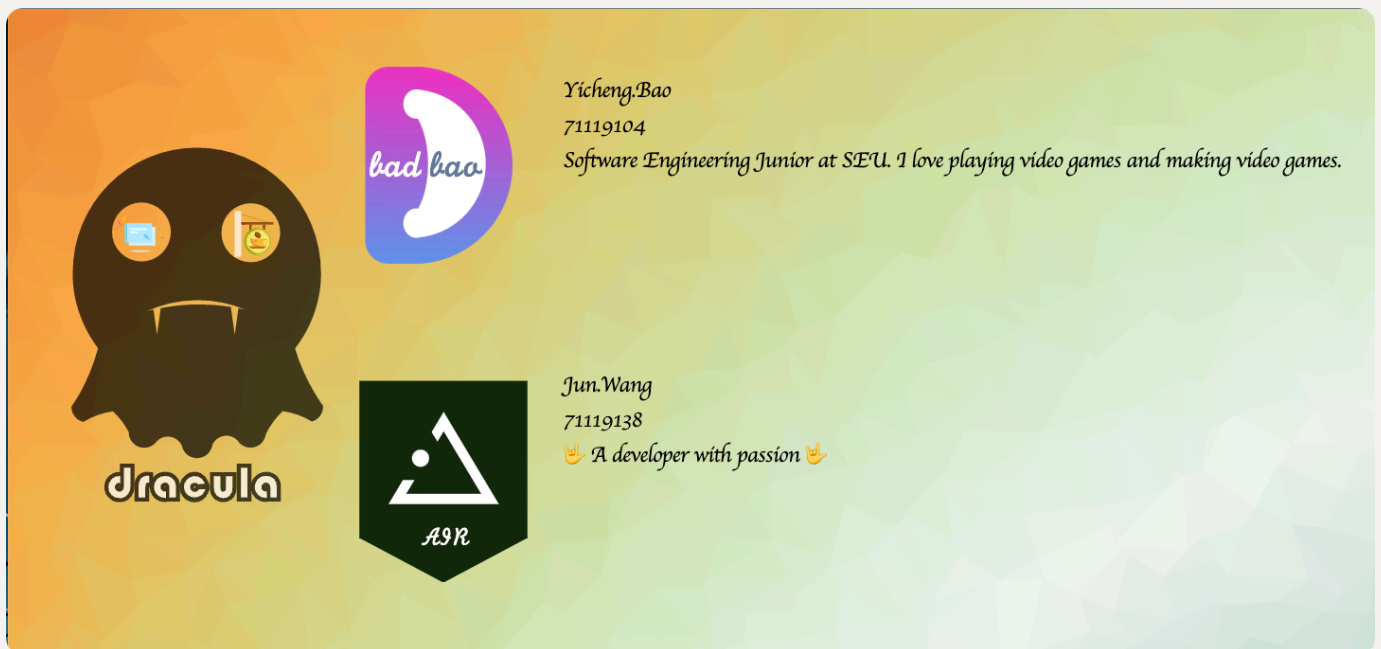
```

        
        <xsl:value-of select="advertises/email" />
    </div>
    <div style="text-align:center; font-family:cursive; font-
size:35px;margin-top:20px">
        
        <xsl:value-of select="advertises/github" />
    </div>
</div> -->
</body>
</html>
</xsl:template>

</xsl:stylesheet>

```

3.效果截图





male

name	country	age	ranking
Boyang JIN	China	24	9
Yuzuru HANYU	Japan	27	4
Nathan CHEN	USA	22	1
Yuma KAGIYAMA	Japan	18	2
Morisi KVITELASHVILI	Georgia	26	10

female

name	country	age	ranking
Anna SHCHERBAKOVA	Russia	17	1
Kaori SAKAMOTO	Japan	21	3
Kamila VALIEVA	Russia	15	4
Alysa LIU	USA	16	7
Loena HENDRICKX	Belgium	22	8

Mixed Pair

ranking	country	members	scores
1	China	Wenjing SUI Cong HAN	9 8 9



thisisbadbao@gmail.com



github.com/thisisbadBao



thoughts.times@gmail.com



github.com/acupofair



dracula

实验任务2_2

1.任务描述

(1) 扩展XML信息模型

扩展数据类型，满足查询需求，例如按名称、时间、日期、数字、数值等数据需求的查询。

(2) 利用XQuery技术，编写XQuery查询语句，按小组，对案例中所有数据，完成以下查询需求

[1]自行选择两种不同的节点，分别进行分类排序，且选择前2条数据和后3条数据，输出为新的xml，文档qs1.xml, qs2.xml。

设计分类排序网页，显示qs1.xml，qs2.xml文档数据。建立返回主页的链接。

[2]自行定义两种不同的特殊组合条件，分别进行查询，输出为新的xml文档 qc1.xml, qc2.xml。

设计综合查询网页，显示qc1.xml和qc2.xml文档数据。建立返回主页的链接。

[3]对[1]和[2]的4个查询结果文档，新设计一个信息模型，进行部分选择与合并，重新构建为一个新的xml文档query.xml。

按新的信息模型，设计合并网页，显示query.xml文档数据。

2.主要实验代码（XQuery代码）

- qc1.xqy

```
for $x in doc("dataset.xml")/figureSkating/singleSki/contestant
where $x/age < 20 and $x/ranking < 7
order by $x/ranking
return $x
```

- qc2.xqy

```
for $x in doc("dataset.xml")/figureSkating/mixedPair/team
where $x/ranking < 7 and $x/womanPlayer/nationality = 'USA'
order by $x/ranking
return $x
```

- qs1.xqy

```
(::~ 筛选排名前2的单人滑雪女参赛者的姓名与年纪 ::)
for $x in doc("dataset.xml")/figureSkating/singleSki/contestant
order by $x/ranking
return <player>{$x/ranking} {$x/name} {$x/age}</player>
(::~ 由于一条记录刚好占五行，所以输出重定向时直接过管道 head即可，head默认读前10行 ::)
(::~ basex qs1.xqy | head > qs1.xml ::)
```

- qs2.xqy

```
for $x in doc("dataset.xml")/figureSkating/mixedPair/team/manPlayer
order by $x/age
return <player> {$x/name} {$x/age}</player>
```

3.效果截图





Name:Vladimir MOROZOV

Age:22

Name:Cong HAN

Age:29

Name:Brandon FRAZIER

Age:39



dracula



Ranking:6

Scores:7 8 7

Name:Alexa Knierim Name:Brandon Frazier

Country:USA

Country:USA

Age:30

Age:39



Dracula



Ranking:1

Name:Anna SHCHERBAKOVA

Age:17

Ranking:2

Name:Yuma KAGIYAMA

Age:18

Name:Vladimir MOROZOV

Age:22

Name:Cong HAN

Age:29

Name:Brandon FRAZIER

Age:39

