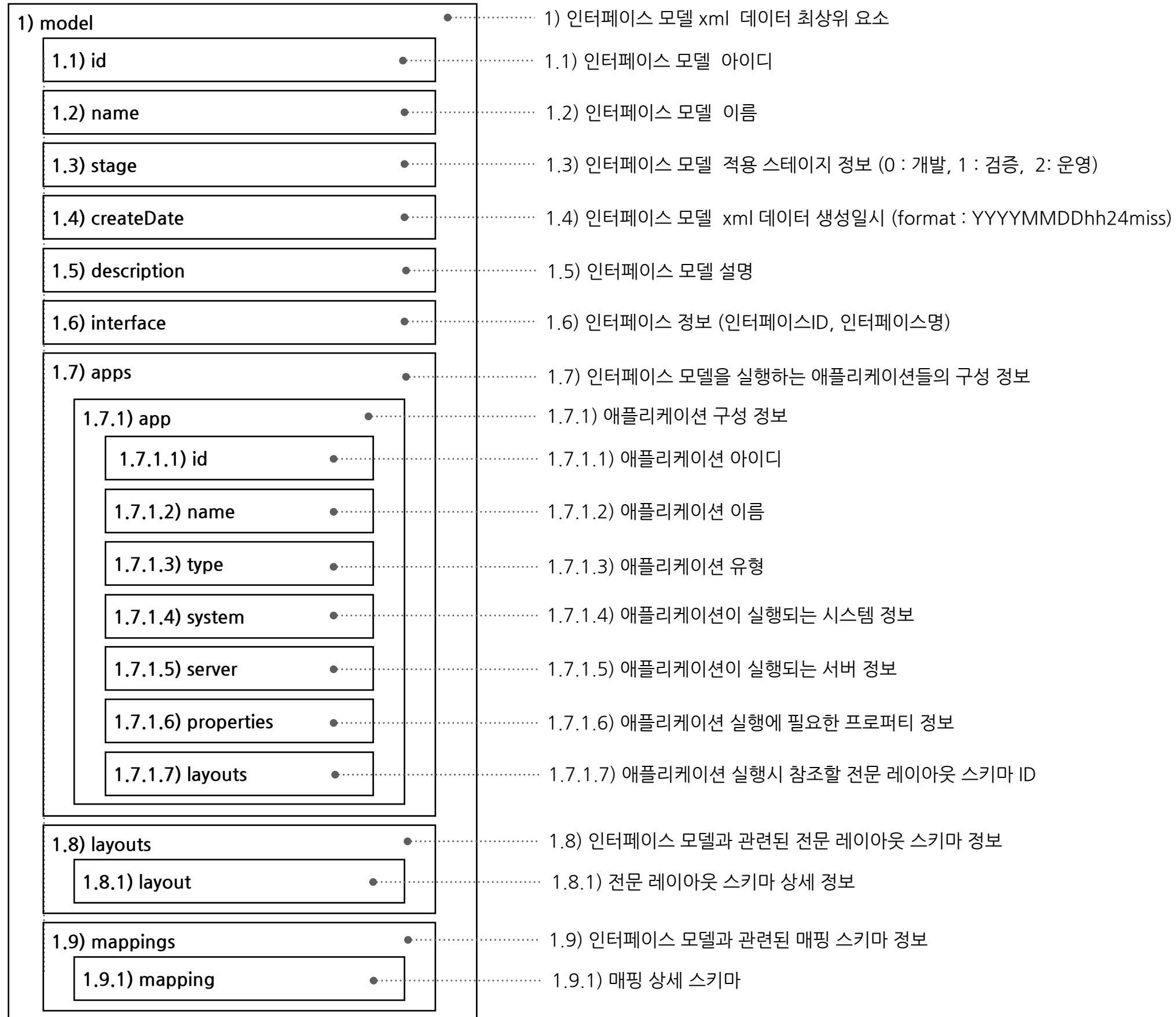


인터페이스 모델 XML 구조

표준 배포 포맷 XML, XSD

1 인터페이스모델 XML 배포 데이터 구조

1 인터페이스모델 XML 배포 데이터 구조



1 인터페이스모델 XML 배포 데이터 구조 - interface

1) model

1.1) id

1.2) name

1.3) stage

1.4) createDate

1.5) description

1.6) interface

1.7) apps

1.7.1) app

1.7.1.1) id

1.7.1.2) name

1.7.1.3) type

1.7.1.4) system

1.7.1.5) server

1.7.1.6) properties

1.7.1.7) layouts

1.8) layouts

1.8.1) layout

1.9) mappings

1.9.1) mapping

1.6) interface

attribute

id : 인터페이스 ID

name : 인터페이스 명

element

business : 업무
attribute (cd, description, name)

channel : 연계솔루션
attribute (cd, description, name)

dataPrDir : Data 처리 방향
attribute (cd, description, name)

dataPrMethod : Data 처리 방식
attribute (cd, description, name)

dataSeq : Data 순차보장 구분
attribute (value[Y|N], description)

dataFrequency : Data 발생주기
attribute (cd, description, name, value,)

sizePerTran : 건별 Data 크기
attribute (description, value)

cntPerFrequency : 주기별 건수
attribute (description, value)

cntPerDay : 일일 발생회수
attribute (description, value)

qttPerDay : 일전송량
attribute (description, value)

1 인터페이스모델 XML 배포 데이터 구조 - layout

1) model

1.1) id

1.2) name

1.3) stage

1.4) createDate

1.5) description

1.6) interface

1.7) apps

1.7.1) app

1.7.1.1) id

1.7.1.2) name

1.7.1.3) type

1.7.1.4) system

1.7.1.5) server

1.7.1.6) properties

1.7.1.7) layouts

1.8) layouts

1.8.1) layout

1.9) mappings

1.9.1) mapping

1.8.1) layout

attribute

id : 레이아웃 CD

modDate : 수정일시 (yyyymmddHH24MISS)

modUserId : 수정자 ID

name : 레이아웃 명

regDate : 등록일시 (yyyymmddHH24MISS)

regUserId : 등록자 ID

element

<![CDATA[...]]>레이아웃 XML Contents

1 인터페이스모델 XML 배포 데이터 구조 - mapping

1) model

1.1) id

1.2) name

1.3) stage

1.4) createDate

1.5) description

1.6) interface

1.7) apps

1.7.1) app

1.7.1.1) id

1.7.1.2) name

1.7.1.3) type

1.7.1.4) system

1.7.1.5) server

1.7.1.6) properties

1.7.1.7) layouts

1.8) layouts

1.8.1) layout

1.9) mappings

1.9.1) mapping

1.9.1) mapping

attribute

id : 매핑 CD

ioType : 매핑 용도 구분 (0: 요청 매핑, 1: 응답 매핑)

modDate : 수정일시 (yyyymmddHH24MISS)

modUserId : 수정자 ID

name : 매핑 명

regDate : 등록일시 (yyyymmddHH24MISS)

regUserId : 등록자 ID

element

<![CDATA[...]]>매핑 XML Contents

2 인터페이스모델 XML 배포 데이터 예시

2 인터페이스모델 XML 배포 데이터 예시

```
<?xml version="1.0" encoding="UTF-8"?>
<model xmlns="http://www.w3.org/2001/XMLSchema-instance">
  <id>103</id>
  <name>배포명2</name>
  <stage>1</stage>
  <createDate>20201109080442</createDate>
  <description>배포명2</description>
  <interface id="SAMPLE-001" name="UI-FW 샘플"/>
```

인터페이스모델

애플리케이션

```
<apps>
  <app id="1" name="Vitria DB Service">
    <type cd="VDS" name="Vitria DB Service"/>
    <system cd="EH" name="본부" seq="0" type="0"/>
    <server cd="BO1" name="BOS 1"/>
    <properties>
      <property name="isProcedure">TRUE</property>
      <property name="sql"><![CDATA[select * from TB_USER]]></property>
      <property name="sqlType">prepared sql</property>
      <property name="tableName">TB_USER</property>
      <property name="crudType">select</property>
      <property name="serviceName">DB서비스명11</property>
    </properties>
    <layouts>
      <layout ref="R001" type="1"/>
    </layouts>
  </app>
  <app id="2" name="Vitria DB Service">
    <type cd="VDS" name="Vitria DB Service"/>
    <system cd="BO" name="BOS" seq="1" type="2"/>
    <server cd="BO2" name="BOS 2"/>
    <properties>
      <property name="isProcedure">FALSE</property>
      <property name="sql"><![CDATA[SQL22]]></property>
      <property name="sqlType">literal sql</property>
      <property name="tableName">테이블명22</property>
      <property name="crudType">insert</property>
      <property name="serviceName">DB서비스명22</property>
    </properties>
    <layouts>
      <layout ref="S001" type="1"/>
    </layouts>
  </app>
</apps>
```

전문 레이아웃 스키마

```
<layouts>
  <layout id="R001" name="수신레이아웃1">
    <![CDATA[<?xml version="1.0" encoding="UTF-8"?><xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      ...
    </xs:schema>]]>
  </layout>
  <layout id="S001" name="송신레이아웃1">
    <![CDATA[<?xml version="1.0" encoding="UTF-8"?><xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      ...
    </xs:schema>]]>
  </layout>
</layouts>
```

상세 전문 레이아웃 스키마 정보 (CDATA 로 포함)

매핑 스키마

```
<mappings>
  <mapping id="SRMAP001" ioType="0" name="송수신매핑1">
    <![CDATA[<?xml version="1.0" encoding="UTF-8"?><map xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="map.xsd">
      ...
    </map>]]>
  </mapping>
</mappings>
```

상세 매핑 스키마 정보 (CDATA 로 포함)

```
</model>
```


3 전문 레이아웃 스키마

3 전문 레이아웃 스키마

3.1 전문 레이아웃 스키마 구조

전문 레이아웃을 정의하기 위해 본 시스템에서는 XML 문서의 구조 및 해당 문서가 포함할 수 있는 적법한 요소와 속성을 명시하기 위한 목적의 표준 표기법인 XSD(XML Schema Definition) 를 사용합니다. (2001년 W3C 제정 표준 스키마 정의 언어)

본 문서에서 사용한 표기 외에 추가적으로 더 상세한 XSD 표기법에 대한 정보는 시중의 레퍼런스 사이트를 찾아 보시기 바랍니다.

```
1: <?xml version="1.0" encoding="UTF-8"?>
2: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
3:   <xs:element name="cusAccountInfoDto">
4:     <xs:annotation>
5:       <xs:documentation>고객계좌정보조회</xs:documentation>
6:       <xs:appinfo>
7:         <mx:properties xmlns:mx="http://mint.mocomsys.com/2020/schema">
8:           <mx:property key="format" value="xml"/>
9:         </mx:properties>
10:      </xs:appinfo>
11:    </xs:annotation>
12:    <xs:complexType>
13:      <xs:sequence>
14:        <xs:element maxOccurs="1" minOccurs="1" name="custId">
15:          <xs:simpleType>
16:            <xs:annotation>
17:              <xs:documentation>고객ID</xs:documentation>
18:            </xs:annotation>
19:            <xs:restriction base="xs:string">
20:              <xs:maxLength value="10"/>
21:            </xs:restriction>
22:          </xs:simpleType>
23:        </xs:element>
24:        ...
25:        <xs:element maxOccurs="1" minOccurs="1" name="accountTrxInfoDto">
26:          <xs:annotation>
27:            <xs:documentation>계좌거래정보</xs:documentation>
28:          </xs:annotation>
29:          <xs:complexType>
30:            <xs:sequence>
31:              <xs:element maxOccurs="1" minOccurs="1" name="accNum">
32:                <xs:simpleType>
33:                  <xs:annotation>
34:                    <xs:documentation>계좌번호</xs:documentation>
35:                  </xs:annotation>
36:                  <xs:restriction base="xs:string">
37:                    <xs:maxLength value="6"/>
38:                  </xs:restriction>
39:                </xs:simpleType>
```

[표기법 설명을 위한 XSD]

[사용된 XSD 표기법 설명]

xmlns:xs 속성은 XSD의 요소와 타입에 사용할 W3C의 XML 스키마 네임스페이스를 명시합니다.

element 는 xml 문서 내의 데이터 요소를 정의하는 단위로 name을 부여할 수 있습니다.

annotation 는 element 의 추가적인 설명을 정의하기 위한 용도로 사용되며 문장을 통한 설명인 **documentation** 정보와 사용자가 정의한 xml 형태의 정보를 하위에 추가 정의 가능한 **appinfo** 로 구성됩니다.

complexType 는 element 의 타입중 하나로 복수의 element를 모아 복합적인 하나의 새로운 집합 element로 표현할 수 있는 복합 유형 표기를 위한 타입입니다.

sequence 는 complexType 내의 element 가 표기 순서대로 표현되도록 제한합니다.

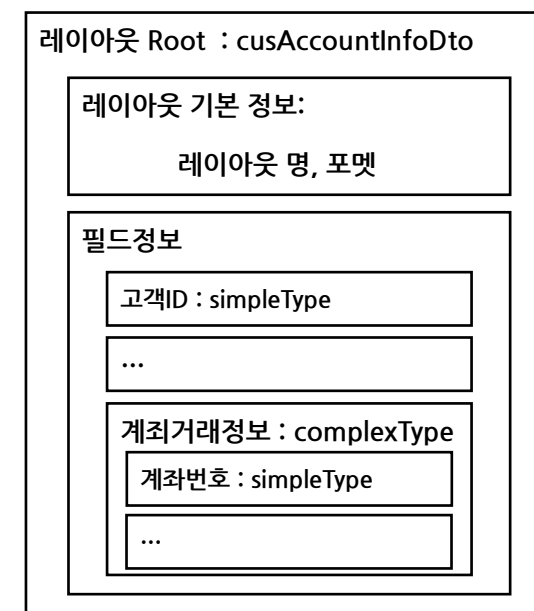
maxOccurs 는 element가 데이터 내에서 최대 몇 회 발생될 지 제한합니다.

minOccurs 는 element가 데이터 내에서 최소 몇 회 발생될 지 제한합니다.

simpleType 은 complexType과 달리 하나의 element에 대해 정의하기 위한 사용됩니다.

simpleType 은 **restriction base** 를 통해 데이터 유형을 제한할 수 있으며, 데이터 타입이 숫자라면 **totalDigits** 정의를 통해 숫자의 길이를 제한할 수 있습니다.

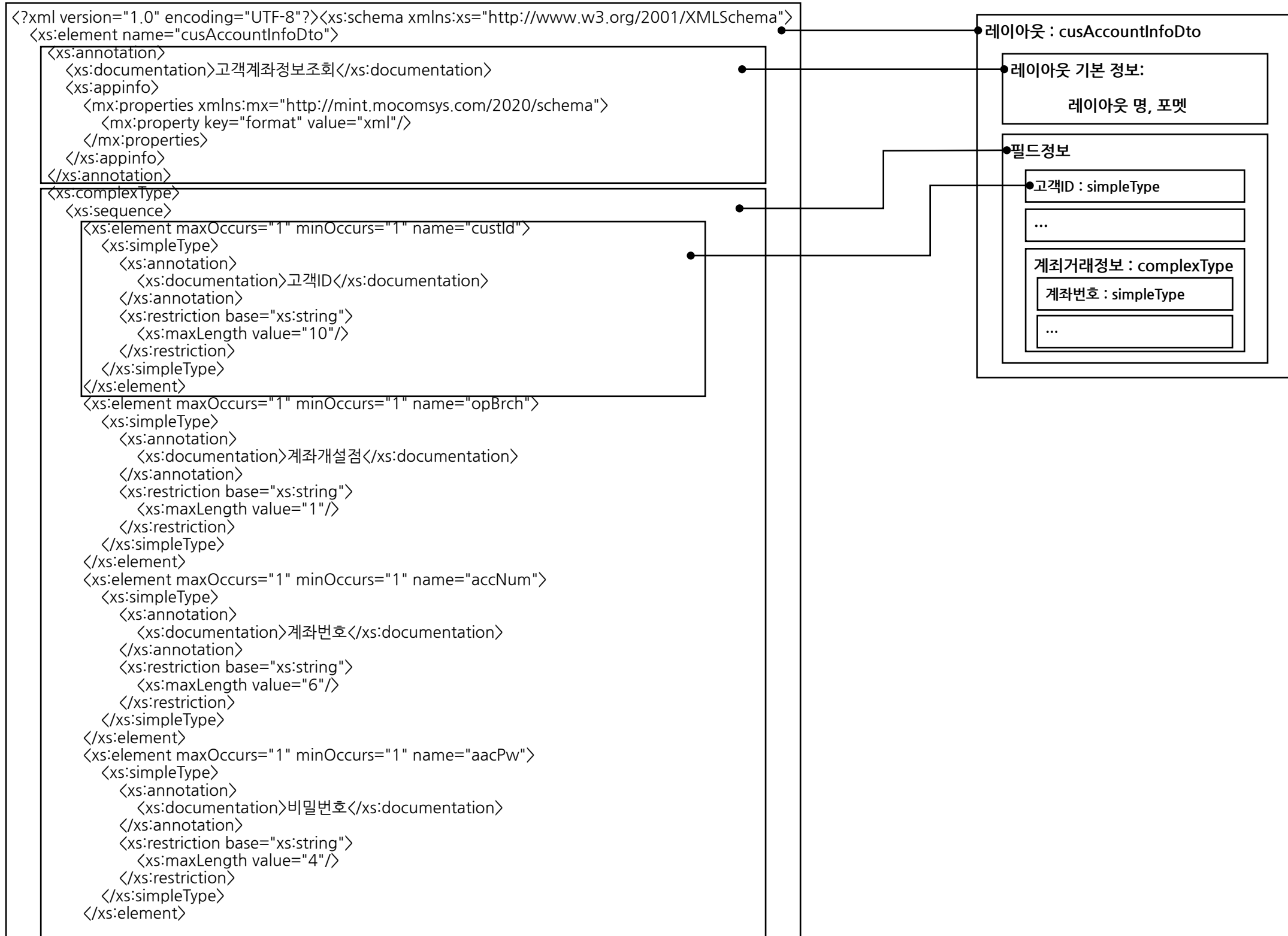
restriction base 가 string 일 경우 **maxLength**를 통해 데이터 문자열의 최대 길이를 제한할 수 있습니다.



[전문 레이아웃 XSD 구조 예]

3 전문 레이아웃 스키마

3.2 전문 레이아웃 스키마 작성 예시(1/3)



3.2 전문 레이아웃 스키마 작성 예시(2/3)

```

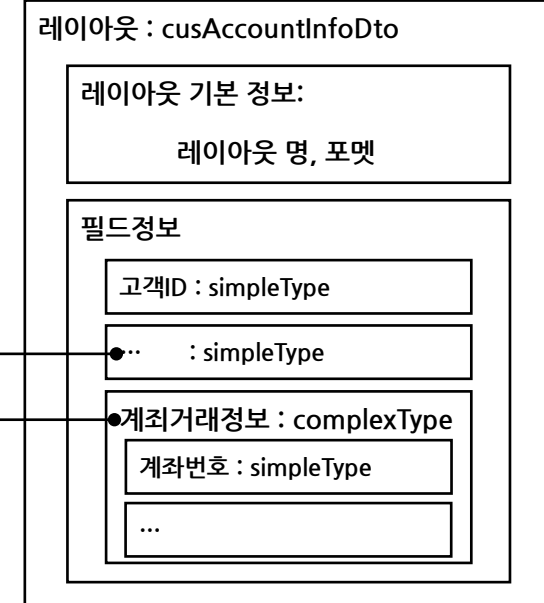
<xs:element maxOccurs="1" minOccurs="1" name="aacPw">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>비밀번호</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element maxOccurs="1" minOccurs="1" name="amt">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>잔액</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element maxOccurs="1" minOccurs="1" name="cusNm">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>고객명</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

```

```

<xs:element maxOccurs="1" minOccurs="1" name="accountTrxInfoDto">
  <xs:annotation>
    <xs:documentation>계좌거래정보</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="1" minOccurs="1" name="accNum">
        <xs:simpleType>
          <xs:annotation>
            <xs:documentation>계좌번호</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:string">
            <xs:maxLength value="6"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element maxOccurs="1" minOccurs="1" name="trxDate">
        <xs:simpleType>
          <xs:annotation>
            <xs:documentation>거래일자</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:string">
            <xs:maxLength value="8"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```



3.2 전문 레이아웃 스키마 작성 예시(3/3)

```

<xs:simpleType>
  <xs:annotation>
    <xs:documentation>거래시각</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:maxLength value="6"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element maxOccurs="1" minOccurs="1" name="trxAmt">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>거래금액</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element maxOccurs="1" minOccurs="1" name="trxType">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>거래유형명</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="6"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

레이아웃 : cusAccountInfoDto

레이아웃 기본 정보:

레이아웃 명, 포맷

필드정보

고객ID : simpleType

... : simpleType

●계좌거래정보 : complexType

계좌번호 : simpleType

...

3 전문 레이아웃 스키마

3.3 전문 레이아웃 스키마 ELEMENT 설명

3.3.1 전문 레이아웃 한글명 영문명 포맷 정의

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  • <xs:element name="AABB0001">
    1) <xs:annotation>
      <xs:documentation>고객계좌정보조회</xs:documentation> • 2)
    <xs:appinfo>
      <mx:properties xmlns:mx="http://mint.mocomsys.com/2020/schema">
        <mx:property key="format" value="xml"/>
      </mx:properties>
      3)
    </xs:appinfo>
  </xs:element>
</xs:schema>
```

- 1) 최상위 Root Element 는 전문 레이아웃 등록 시 입력한 코드(CD)값을 Element name 값으로 사용합니다.
- 2) 최상위 Root Element 에 속한 **documentation** 값은 전문 레이아웃 등록시 입력한 한글명 정보입니다.
- 3) 최상위 Root Element 에 속한 **appinfo** 의 **property** 의 **key, value** 인 **format, xml** 값은 전문 레이아웃 등록시 입력한 유형 값에 해당합니다.

레이아웃 상세 정보

수정

삭제

이전화면

코드(CD)

AABB0001

한글명

고객계좌정보조회

영문명

cusAccountInfoDto

구분

레이아웃

복합유형

유형

xml

길이

107

암호화

없음

등록자

포탈관리자

등록일

2020/11/10

레이아웃 필드 정보

명세서/개정이력

엑셀 저장

필드명(영)(*)	필드명(한)(*)	유형(*)	오프셋	길이	기본값	암호화	길이필드구분	반복필드참조	반복필드	반복횟수	정렬기준	채움문자
custId	고객ID	string	0	10			N	N		0	왼쪽정렬	
opBrch	계좌개설점	string	10	1			N	N		0	왼쪽정렬	
accNum	계좌번호	string	11	6			N	N		0	왼쪽정렬	
aacPw	비밀번호	string	17	4			N	N		0	왼쪽정렬	
amt	잔액	string	21	20			N	N		0	왼쪽정렬	
cusNm	고객명	string	41	20			N	N		0	왼쪽정렬	
accountTrxInfoDto	계좌거래정보	complex	0	0			N	N		0	왼쪽정렬	
accNum	계좌번호	string	61	6			N	N		0	왼쪽정렬	
trxDate	거래일자	string	67	8			N	N		0	왼쪽정렬	
trxTime	거래시각	string	75	6			N	N		0	왼쪽정렬	
trxAmt	거래금액	string	81	20			N	N		0	왼쪽정렬	
trxType	거래유형명	string	101	6			N	N		0	왼쪽정렬	

3 전문 레이아웃 스키마

3.3 전문 레이아웃 스키마 ELEMENT 설명

3.3.2 단순 유형 필드 정의

```
<xs:element maxOccurs="1" minOccurs="1" name="cusNm">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>고객명</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

1) 필드 cusNm 의 영문명을 element 의 name으로 정의하고 반복 회수 "1" 을 지정합니다.

2) 필드 cusNm 의 한글명을 정의합니다.

3) 필드 cusNm 의 데이터 유형을 string 으로 정의합니다.

4) 필드 cusNm 의 데이터 길이를 20으로 정의합니다.

레이아웃 상세 정보

[수정](#)[삭제](#)[이전화면](#)

코드(CD)

AABB0001

한글명

고객계좌정보조회

영문명

cusAccountInfoDto

구분

☒ 레이아웃 ☐ 복합유형

유형

xml

길이

107

암호화

없음

등록자

포탈관리자

등록일

2020/11/10

레이아웃 필드 정보

[명세서/개정이력](#)

엑셀 저장

필드명(영)(*)	필드명(한)(*)	유형(*)	오프셋	길이	기본값	암호화	길이필드구분	반복필드참조	반복필드	반복횟수	정렬기준	채움문자
custId	고객ID	string	0	10			N	N		0	왼쪽정렬	
opBrch	계좌개설점	string	10	1			N	N		0	왼쪽정렬	
accNum	계좌번호	string	11	6			N	N		0	왼쪽정렬	
aacPw	비밀번호	string	17	4			N	N		0	왼쪽정렬	
amt	잔액	string	21	20			N	N		0	왼쪽정렬	
cusNm	고객명	string	41	20			N	N		0	왼쪽정렬	
accountTrxInfoDto	계좌거래정보	complex	0	0			N	N		0	왼쪽정렬	
accNum	계좌번호	string	61	6			N	N		0	왼쪽정렬	
trxDate	거래일자	string	67	8			N	N		0	왼쪽정렬	
trxTime	거래시각	string	75	6			N	N		0	왼쪽정렬	
trxAmt	거래금액	string	81	20			N	N		0	왼쪽정렬	
trxType	거래유형명	string	101	6			N	N		0	왼쪽정렬	

3 전문 레이아웃 스키마

3.3 전문 레이아웃 스키마 ELEMENT 설명

3.3.3 복합 유형 필드 정의



- 1) 복합유형 필드 “accountTrxInfoDto 계좌거래정보” 정의를 시작부분입니다. element의 name을 영문명으로 지정합니다.
- 2) 복합유형 필드 “accountTrxInfoDto”의 한글명을 정의합니다.
- 3) 복합유형 필드 “accountTrxInfoDto”의 첫번째 필드를 정의 합니다. 필드 영문명을 accNum으로 정의하고 반복은 없는 적으로 정의 합니다.
- 4) 필드 accNum의 한글명을 계좌번호로 정의합니다.
- 5) 필드 accNum의 데이터 유형을 string으로 정의합니다.
- 6) 필드 accNum의 길이를 6으로 정의합니다.
- 7) 복합유형 “accountTrxInfoDto 계좌거래정보”의 두 번째 단순유형 필드를 정의합니다.

필드명(영)(*)	필드명(한)(*)	유형(*)	오프셋	길이	기본값	암호화	길이필드구분	반복필드참조	반복필드	반복횟수	정렬기준	채움문자
custId	고객ID	string	0	10			N	N		0	왼쪽정렬	
opBrch	계좌개설점	string	10	1			N	N		0	왼쪽정렬	
accNum	계좌번호	string	11	6			N	N		0	왼쪽정렬	
aacPw	비밀번호	string	17	4			N	N		0	왼쪽정렬	
amt	잔액	string	21	20			N	N		0	왼쪽정렬	
cusNm	고객명	string	41	20			N	N		0	왼쪽정렬	
▲ accountTrxInfoDto	계좌거래정보	complex	0	0			N	N		0	왼쪽정렬	
accNum	계좌번호	string	61	6			N	N		0	왼쪽정렬	
trxDate	거래일자	string	67	8			N	N		0	왼쪽정렬	
trxTime	거래시각	string	75	6			N	N		0	왼쪽정렬	
trxAmt	거래금액	string	81	20			N	N		0	왼쪽정렬	
trxType	거래유형명	string	101	6			N	N		0	왼쪽정렬	

3 전문 레이아웃 스키마

3.3 전문 레이아웃 스키마 ELEMENT 설명

3.3.4 필드 속성 정의

- 1) 전문 레이아웃 필드의 기본 속성 외의 추가적인 속성 정의는 정의된 값이 없을 경우 표현하지 않는 것을 원칙으로 합니다.
- 2) 패딩 값이 정의되지 않았을 경우 데이터 유형이 문자일 경우 공백(" ") 문자열을 숫자일 경우 "0" 값을 패딩으로 처리하는 것으로 합니다.
- 3) 정렬 값이 정의되지 않았을 경우 데이터 유형이 문자일 경우 오른쪽 정렬(right), 숫자일 경우 왼쪽 정렬(left)로 처리하는 것으로 합니다.
- 4) 필드 속성 정의 방법

```
<xs:element maxOccurs="1" minOccurs="1" name="accNum">
  <xs:simpleType>
    <xs:annotation>
      <xs:documentation>계좌번호</xs:documentation>
```

```
    <xs:appinfo>
```

```
      <mx:properties xmlns:mx="http://mint.mocomsys.com/2020/schema">
```

```
        <mx:property key="required" value="yes"/>
```

<!-- 1) 필수 여부(yes | no), 기본 값 : yes -->

```
        <mx:property key="scale" value="0"/>
```

<!-- 2) 스케일, 기본 값 : 0 -->

```
        <mx:property key="justify" value="right"/>
```

<!-- 3) 정렬방식, 정의하지 않을 경우 데이터 유형이 문자일 경우 오른쪽 정렬(right), 숫자일 경우 왼쪽 정렬(left)로 간주합니다.-->

```
        <mx:property key="padding" value="0"/>
```

<!-- 4) 채움문자, "space"는 공백 문자(" ")를 의미합니다. 정의하지 않을 경우 데이터 유형이 문자일 경우 공백(" ") 으로 숫자일 경우 "0" 으로 간주합니다.-->

```
        <mx:property key="isKorean" value="no"/>
```

<!-- 5) 한글여부(yes | no), 기본 값 : no -->

```
        <mx:property key="meta" value="yes"/>
```

<!-- 6) 메타체크여부(yes | no), 기본 값 : yes -->

```
        <mx:property key="decimalPoint" value="yes"/>
```

<!-- 7) 소수점포함여부(yes | no), 기본 값: yes -->

```
      </mx:properties>
```

```
    </xs:appinfo>
```

```
  </xs:annotation>
```

```
  <xs:restriction base="xs:string">
```

```
    <xs:maxLength value="6"/>
```

```
  </xs:restriction>
```

```
</xs:simpleType>
```

```
</xs:element>
```

“계좌번호” 필드 속성 정의

* 위 항목 외에 필드 추가 속성이 발생하면 appinfo property 항목의 key, value 값 형식으로 적용됩니다.

4 전문 매핑 스키마

4 전문 매핑 스키마

4.1 전문 매핑 구조

입력과 출력 전문 간 매핑을 정의하기 위해 본 시스템에서는 map.xsd 를 통해 스키마를 정의합니다.

map.xsd 는 다음에서 참조 가능합니다.

`/mint-endpoint/src/main/resources/config/map.xsd`

아래 매핑 XML 예시는 map.xsd 정의에 따라 작성된 두 전문간의 매핑 내용입니다.

```
1: <?xml version="1.0" encoding="UTF-8"?><map xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="map.xsd">
2:   <description>
3:     <name>응답전문매핑002</name>
4:     <cd>RLYMAP002</cd>
5:   </description>
6:   <dataset>
7:     <source id="NCS12345_I_1"/>
8:     <target id="NCS12345_O_1"/>
9:   </dataset>
10:  <mapping>
11:    <item>
12:      <control id="move"/>
13:      <input>
14:        <field dataset="NCS12345_I_1" path="/cusAccountInfoDto"/>
15:      </input>
16:      <output>
17:        <field dataset="NCS12345_O_1" path="/cusAccountInfoDto"/>
18:      </output>
19:    </item>
20:    <item>
21:      <control id="move"/>
22:      <input>
23:        <field dataset="NCS12345_I_1" path="/cusAccountInfoDto/CUST_ID"/>
24:      </input>
25:      <output>
26:        <field dataset="NCS12345_O_1" path="/cusAccountInfoDto/CUST_ID"/>
27:      </output>
28:    </item>
29:    <item>
30:      <control id="move"/>
31:      <input>
32:        <field dataset="NCS12345_I_1" path="/cusAccountInfoDto/CUS_NM"/>
33:      </input>
34:      <output>
35:        <field dataset="NCS12345_O_1" path="/cusAccountInfoDto/CUS_NM"/>
36:      </output>
37:    </item>
38:  </mapping>
39: </map>
```

입력 전문 “NCS12345_I_1” 의 필드 “cusAccountInfoDto” 를
출력 전문 “NCS12345_O_1” 의 필드 “cusAccountInfoDto” 로
매핑한다는 의미로 복합유형(ComplexType)의 하위 필드가 매핑에
포함되면 기본적으로 최상위 복합유형(ComplexType) 필드는
매핑 item에 포함됩니다.

입력 전문 “NCS12345_I_1” 의 필드 “cusAccountInfoDto.CUST_ID” 를
출력 전문 “NCS12345_O_1” 의 필드 “cusAccountInfoDto.CUST_ID” 로
매핑한다는 의미입니다.

[매핑 XML 예시]