

Xpath fonksiyon Örnekleri

Tekrar eden kaytlar silme - VKN si ayn olan kaytlar siler.

```
<Group Description="Tekrar eden kaytlar sil">
  <Delete Node="/*/Liste/Data/Template[VKN=preceding::VKN]" />
</Group>
```

Bir Node Set içerisinde distinct görevi görür - Tip Deeri için

```
/*/IK_Tanimlamalar/Data/Template[not(Tip=preceding::Tip)]/Tip
```

Süreç içerisinde listelenmi birden çok satrdaki alan tek bir satrda göstermek amac ile ekleme yaparken ayn isimde olanlar belirlemek için xpath **contains** özelliği kullanılabilir.

Contains ve string-length kullanm

```
<ForEach Node="/*/FirmaBilgileri/Liste/Sablon[@guid!='{00000000-0000-0000-0000-000000000000}' ]">
  <If>
    <Then>
      <Condition>
        <Not>
          <Value Culture="" Target="None">eval::contains(//*[FormBilgileri/FirmaAd,FirmaAdi]</Value>
        </Not>
      </Condition>
      <Perform>
        <Set Node="//*[FormBilgileri/FirmaAd]">
          <StringConcat>
            <Evaluate Target="None" Culture=""><![CDATA[//*[FormBilgileri/FirmaAd]]></Evaluate>
            <Value Culture="" Target="None">-</Value>
            <Value Culture="" Target="None">eval::../FirmaAdi</Value>
          </StringConcat>
        </Set>
      </Perform>
    </Then>
  </If>
</ForEach>
<SetVariable Name="$(firstLength)">
  <Math Operator="Subtract">
    <Value Culture="" Target="None">eval::string-length(//*[FormBilgileri/FirmaAd]</Value>
    <Value Culture="" Target="Integer">1</Value>
  </Math>
</SetVariable>
<Set Node="//*[FormBilgileri/FirmaAd]">
  <SubString StartIndex="1" Length="$(firstLength)">
    <Evaluate Target="None" Culture=""><![CDATA[//*[FormBilgileri/FirmaAd]]></Evaluate>
  </SubString>
</Set>
```

preceding-sibling kullanm

```
<Script Name="">
  <SetVariable Name="$(Bilgilendirilecekler)">
    <Value Culture="" Target="None" />
  </SetVariable>
  <ForEach Node="/*/processingInfo/workItem[not(completedBy = preceding-sibling::workItem/completedBy)]">
    <SetVariable Name="$(Bilgilendirilecekler)">
      <Value Culture="" Target="None">eval::concat('${Bilgilendirilecekler}', completedBy, ';')</Value>
    </SetVariable>
  </ForEach>
  <SetVariable Name="$(length)">
    <Math Operator="Subtract">
      <Value Culture="" Target="None">eval::string-length('${Bilgilendirilecekler}')</Value>
      <Value Culture="" Target="Integer">1</Value>
    </Math>
  </SetVariable>
  <SetVariable Name="$(Bilgilendirilecekler)">
    <Value Culture="" Target="None">eval::substring('${Bilgilendirilecekler}',1,'$(length)')</Value>
  </SetVariable>
  <Value Culture="" Target="None">$(Bilgilendirilecekler)</Value>
</Script>
```

floor ve div ile bölme ilemi kullanm

```
<Set Node="Sure/Gun">
  <Evaluate Target="Decimal" Culture=""><![CDATA[floor(Sure/Saat div 24)]]</Evaluate>
</Set>
```

Bölüm sonras kalan alma - mod

```
<Set Node="Sure/Saat">
  <Evaluate Target="Decimal" Culture=""><![CDATA[Sure/Saat mod 24]]</Evaluate>
</Set>
```

Lower-Case

```
<Evaluate Target="None" Culture=""><![CDATA[translate(/*/Baslik
/Email, 'ABCDEFGHJKLMNOPQRSTUVWXYZ', 'abcdefghijklmnopqrstuvwxyz')]]</Evaluate>
```

Upper-Case

```
<Evaluate Target="None" Culture=""><![CDATA[translate(.
/Kod, 'abcdefghijklmnopqrstuvwxyzçöüüâ', 'ABCDEFGHIJKLMNOPQRSTUVWXYZCCGGIIIOSSUU ')]></Evaluate>
```

String-join

```
<Value Culture="" Target="None">eval::string-join((/*/Tanimlamalar/Data/Template[(Kod='823' or Kod='824' or
Kod='825' or Kod='826' or Kod='827' or Kod='828') and TalepDurumu='true' ]/Ad/text()),"-")</Value>
```

```
<SetVariable Name="$(Hold)">
  <Value Culture="" Target="None">eval::substring-after('$(MailBody)', '/index.html/form/')</Value>
</SetVariable>
<Set Node="workitemid">
  <Value Culture="" Target="None">eval::substring('$(Hold)', 1, 36)</Value>
</Set>
```