

TRANSLATE UYGULAMA KODLARININ AÇIKLANMASI

HTML KODLARI

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <link rel="stylesheet" href="style.css">
  <title>TRANSLATE</title>
</head>

<body>
  <div class="container">
    <div class="wrapper">
      <!-- iki adet metin kutusu oluşturuşdu.-->
      <div class="text-input">
        <!-- "Metin Girin" yazısıyla başlayan bir yer tutucu
içeriyor ve kullanıcıdan bir metin girmesini bekliyor. -->
        <textarea class="from-text" placeholder="Metin
Girin"></textarea>
        <!-- İkinci metin kutusu ise "Çeviri" yazısıyla
başlayan bir yer tutucu içeriyor ve ilk metinde girilen metnin
çevirisini göstermek için kullanılabilir. -->
        <textarea class="to-text"
placeholder="Çeviri"></textarea>
      </div>
      <ul class="controls">
        <li class="row from">
          <!-- İki adet ikonlu buton oluşturun. -->
          <div class="icons">
            <!-- Bir hoparlör simgesidir.Kullanıcının bir
metin parçasının sesli okunmasını sağlamak için kullanılır -->
            <i id="from" class="fas fa-volume-up"></i>
            <!-- Bir kopyala simgesidir. Kullanıcının metin
parçasını panoya kopyalamasını sağlamak için kullanılır -->
            <i id="from" class="fas fa-copy"></i>
          </div>
          <select>

          </select>
        </li>
```

```

        <!-- Bir "değiştir" (exchange) butonudur. -->
        <li class="exchange"><i class="fas fa-exchange-
alt"></i></li>

        <li class="row to">
            <select> </select>
            <div class="icons">
                <!-- Bir kopyala simgesidir. Kullanıcının metin
parçasını panoya kopyalamasını sağlamak için kullanılır -->
                <i id="to" class="fas fa-copy"></i>
                <!-- Bir hoparlör simgesidir.Kullanıcının bir
metin parçasının sesli okunmasını sağlamak için kullanılır -->
                <i id="to" class="fas fa-volume-up"></i>
            </div>
        </li>
    </ul>
</div>

    <!-- Bir "ÇEVİR" butonudur."TRANSLATE" yazısını içeriyor ve
kullanıcının metinleri çevirmek için butona tıklamasını sağlar. -->
    <button>TRANSLATE</button>
</div>

    <!-- Bu kodlar, bir HTML sayfasında bir JavaScript dosyasını
sayfaya dahil etmek için kullanılır. -->
    <script src="countries.js"></script>
    <script src="main.js"></script>
    <script src="https://kit.fontawesome.com/097d750326.js"
crossorigin="anonymous"></script>
</body>
</html>

```

CSS KODLARI

```

/* Bu kod, bir CSS dosyasında kullanılarak Google Fonts hizmetinden
fontların yüklenmesini sağlar. */
@import
url('https://fonts.googleapis.com/css2?family=Crimson+Text:ital,wght@0,
400;0,600;0,700;1,400;1,600;1,700&family=Poppins:wght@500&display=swap'
);
/* Bu kod, bir CSS dosyasında kullanılarak tüm HTML öğelerinin
varsayılan özelliklerini sıfırlayan ve kutu modeli özelliğini
düzenleyen kod bloğudur */
*{
    /* margin ve padding özellikleri, tüm HTML öğelerinin kenar
boşluklarını ve dolgularını sıfırlar. */
    margin: 0;

```

```
padding: 0;
/* box-sizing özelliği, kutu modelinde kullanılan hesaplama
yöntemini değiştirir */
box-sizing: border-box;
/* font-family özelliği, tüm HTML öğelerinin yazı tipini "Poppins"
ve yedek olarak "sans-serif" olarak ayarlar */
font-family: 'Poppins','sans-serif';
}
/* Bu kod, bir CSS dosyasında kullanılarak HTML sayfasının gövde
öğesinin stilini düzenler. */
body{
/* display: flex; özelliği, HTML sayfasının gövde öğesini esnek bir
kutulu model olarak ayarlar. Bu, öğelerin sayfanın ortalanması ve
hizalanması için kolay bir yöntem sağlar. */
display: flex;
/* align-items: center; özelliği, esnek kutu modelinde öğelerin
dikey hizalamasını merkezlemek için kullanılır. */
align-items: center;
/* justify-content: center; özelliği, esnek kutu modelinde öğelerin
yatay hizalamasını merkezlemek için kullanılır. */
justify-content: center;
/* min-height: 100vh; özelliği, HTML sayfasının minimum
yüksekliğini tam ekran yüksekliği (viewport height) olarak ayarlar. */
min-height: 100vh;
/* background: #f0cee4; özelliği, sayfa arkaplanının rengini pembe
tonlu bir açık griye ayarlar. */
background: #f0cee4;
}
/* Bu kod, bir CSS dosyasında kullanılarak bir HTML sayfasında yer alan
bir sınıf olan "container" öğesinin stilini düzenler. */
.container{
/* width: 690px; özelliği, "container" öğesinin genişliğini 690
piksel olarak ayarlar. */
width: 690px;
/* background: #e484a8; özelliği, "container" öğesinin arkaplan
rengini pembe tonlu bir kırmızı olarak ayarlar. */
background: #e484a8;
/* border-radius: 7px; özelliği, "container" öğesinin kenarlarının
yuvarlatılmasını 7 piksel olarak ayarlar. */
border-radius: 7px;
/* padding: 30px; özelliği, "container" öğesinin içindeki tüm
içeriğin dış çerçeveye olan uzaklığını 30 piksel olarak ayarlar. */
padding: 30px;
}
/* Bu kod, bir CSS dosyasında kullanılarak bir HTML sayfasında yer alan
bir sınıf olan "wrapper" öğesinin stilini düzenler. */
```

```
.wrapper{
    /* border-radius: 5px; özelliği, "wrapper" öğesinin kenarlarının
yuvarlatılmasını 5 piksel olarak ayarlar. */
    border-radius: 5px;
    /* border:1px solid black; özelliği, "wrapper" öğesinin çerçevesini
1 piksellik siyah bir çizgiyle ayarlar. */
    border:1px solid black;
}
/* Bu kod, bir CSS dosyasında kullanılarak "wrapper" sınıfı içinde yer
alan "text-input" sınıfının stilini düzenler. */
.wrapper .text-input{
    /* display: flex; özelliği, "text-input" öğesinin içeriğinin yatay
olarak hizalanmasını sağlar. */
    display: flex;
    /* border:1px solid black; özelliği, "wrapper" öğesinin
çerçevesini 1 piksellik siyah bir çizgiyle ayarlar. */
    border: 1px solid black;
}
/* Bu kod, bir CSS dosyasında kullanılarak "text-input" sınıfı içinde
yer alan "to-text" sınıfının stilini düzenler. */
.text-input .to-text{
    /* border-radius: 0; özelliği, "to-text" öğesinin kenarlarının
yuvarlatılmasını kaldırır. */
    border-radius: 0;
    /* border:1px solid black; özelliği, "wrapper" öğesinin çerçevesini
1 piksellik siyah bir çizgiyle ayarlar. */
    border-left: 1px solid black;
}
/* Bu kod, bir CSS dosyasında kullanılarak "text-input" sınıfı içinde
yer alan "textarea" öğesinin stilini düzenler. */
.text-input textarea{
    /* height: 250px; özelliği, "textarea" öğesinin yüksekliğini 250
piksel olarak ayarlar. */
    height: 250px;
    /* width: 100%; özelliği, "textarea" öğesinin genişliğini ebeveyn
öğenin genişliğiyle aynı olarak ayarlar. */
    width: 100%;
    /* border: none; özelliği, "textarea" öğesinin çerçevesini
kaldırır.*/
    border: none;
    /*resize: none; özelliği, "textarea" öğesinin yeniden
boyutlandırılmasını engeller. */
    resize: none;
    /* outline: none; özelliği, "textarea" öğesinin çerçeve etrafındaki
varsayılan vurgulamayı kaldırır. */
    outline: none;
}
```

```
    /* font-size: 18px; özelliği, "textarea" öğesinin metin boyutunu 18
piksel olarak ayarlar. */
    font-size: 18px;
    /* border-radius: 5px; özelliği, "textarea" öğesinin kenarlarının
yuvarlatılmasını 5 piksel olarak ayarlar. */
    border-radius: 5px;
    /* background: none; özelliği, "textarea" öğesinin arka plan
rengini kaldırır. */
    background: none;
    /* padding: 10px 15px; özelliği, "textarea" öğesinin içeriğinin sol
ve sağ tarafından 10 piksel, üst ve alt tarafından 15 piksel mesafe
bırakır. */
    padding: 10px 15px;
}
/* Bu CSS kodu, "controls" sınıfına sahip bir liste öğesi (örneğin <ul>
veya <ol>) içindeki öğelerin stilini belirler. */
.controls{
    /* list-style: none; özelliği, liste öğelerinin işaretlerini
kaldırır. */
    list-style: none;
    /* padding: 12px 15px; özelliği, liste öğesi etrafına 12 piksel üst
ve alt, 15 piksel sol ve sağ dolgu ekler. */
    padding: 12px 15px;
}
/* Bu CSS kodu, "controls", "li" ve "icons" sınıflarına sahip öğelerin
stilini belirler. */
.controls,li,.icons{
    /* display: flex; özelliği, öğeleri yatay hizalar. */
    display: flex;
    /* list-style: none; özelliği, liste öğelerinin işaretlerini
kaldırır. */
    list-style: none;
    /* padding: 12px 15px; özelliği, öğelerin etrafına 12 piksel üst ve
alt, 15 piksel sol ve sağ dolgu ekler. */
    padding: 12px 15px;
    /* align-items: center; özelliği, öğeleri dikey yönde ortalar. */
    align-items: center;
    /* justify-content: space-between; özelliği, öğeleri yatay yönde
eşit bir şekilde dağıtır. */
    justify-content: space-between;
}
/* Bu CSS kodu, "controls" sınıfının altındaki "row" sınıfına sahip
"from" sınıfına sahip öğelerin "icons" sınıfındaki stilini belirler. */
.controls .row.from .icons{
    /* padding-right: 10px; özelliği, "icons" sınıfındaki öğelerin sağ
tarafındaki dolguyu 10 piksel olarak belirler. */
```

```
padding-right: 10px;
/* border-right: 1px solid gray; özelliği, "icons" sınıfındaki
öğelerin sağ tarafına 1 piksel genişliğinde gri bir kenarlık ekler. */
border-right: 1px solid gray;
}
/*Bu CSS kodu, "controls" sınıfının altındaki "row" sınıfına sahip "to"
sınıfına sahip öğelerin "icons" sınıfındaki stilini belirler. */
.controls .row.to .icons{
    /* padding-left: 10px; özelliği, "icons" sınıfındaki öğelerin sol
tarafındaki dolguyu 10 piksel olarak belirler. */
padding-left: 10px;
/* border-left: 1px solid gray; özelliği, "icons" sınıfındaki
öğelerin sol tarafına 1 piksel genişliğinde gri bir kenarlık ekler. */
border-left: 1px solid gray;
}
/* Bu CSS kodu, "controls" sınıfının altındaki "icons" sınıfına sahip
öğelerin genişliğini yüzde 38'e ayarlar. */
.controls .icons{
width: 38%;
}
/* Bu CSS kodu, "controls" sınıfının altındaki "icons" sınıfının
altındaki "i" öğelerine bir dizi stil özelliği ekler. */
.controls .icons i{
    /* "width" özelliği, "i" öğelerinin genişliğini 50 piksele ayarlar.
*/
width: 50px;
/* "color" özelliği, "i" öğelerinin yazı rengini siyah olarak
ayarlar. */
color: black;
/* "font-size" özelliği, "i" öğelerinin yazı boyutunu 14 piksel
olarak ayarlar. */
font-size: 14px;
/* "cursor" özelliği, fare imleci "i" öğelerinin üzerine geldiğinde
görüntülenen imleci belirler. */
cursor: pointer;
/* "justify-content" özelliği, "i" öğelerini yatay olarak hizalamak
için kullanılır. */
justify-content: center;
}
/* Seçim kutularının stillerini tanımlamaktadır. */
.controls .row select{
    /* border: none;; Seçim kutusunun sınırını kaldırır. */
border:none;
/* outline: none;; Seçim kutusunun odak çerçevesini kaldırır. */
outline: none;
/* font-size: 18px;; Seçim kutusunun yazı tipi boyutunu ayarlar. */
```

```
font-size: 18px;
/*background: none;; Seçim kutusunun arka planını kaldırır.*/
background: none;
}
/* /* Bu CSS kodu, "controls" sınıfının altındaki "exchange" sınıfının
altındaki "i" öğelerine bir dizi stil özelliği ekler. */ */
.controls .exchange{
    /* "color" özelliği, "exchange" öğelerinin yazı rengini siyah
olarak ayarlar. */
    color:black;
    /* "font-size" özelliği, "exchange" öğelerinin yazı boyutunu 16
piksel olarak ayarlar. */
    font-size: 16px;
    /* "cursor" özelliği, fare imleci "exchange" öğelerinin üzerine
geldiğinde görüntülenen imleci belirler. */
    cursor: pointer;
}
/* Bu CSS kodlar, .container adlı bir HTML öğesinin içindeki tüm button
öğelerine stil özellikleri ekler */
.container button{
    /* width: 100% (yani, öğenin içindeki genişliği kadar) */
    width: 100%;
    /* padding: 14 piksel */
    padding: 14px;
    /*margin-top: 20 piksel */
    margin-top: 20px;
    /* border: hiçbir şey (yani, sınır yok) */
    border: none;
    /* outline: hiçbir şey (yani, dış çizgi yok) */
    outline: none;
    /* cursor: işaretçi şeklinde bir imleç */
    cursor: pointer;
    /*font-size: 20 piksel */
    font-size:20px;
    /* color: beyaz renk (#fff) */
    color: #fff;
    /* border-radius: 10 piksel*/
    border-radius: 10px;
    /*background: siyah renkli arka plan */
    background:black;
}
```

```
/* Responsive */
/* Bu kod bloğu, mobil cihazlarda sayfanın düzgün görünmesi için bazı özellikleri tanımlar. */
@media (max-width:450px){
    /* .container ögesi için genişlik 350 piksel, dolayısıyla bu öge daha küçük olacaktır ve içeriğin daha iyi görüntülenmesini sağlar. Ayrıca padding değeri 10 piksel olarak ayarlanmıştır. */
    .container{
        width: 350px;
        padding: 10px;
    }
    /* .text-input textarea ögesi için yükseklik 200 piksel ve genişlik %50 olarak ayarlanmıştır. Padding değeri de 10 piksel yükseklik, 15 piksel genişlik olarak ayarlanmıştır. */
    .text-input textarea{
        height: 200px;
        width: 50%;
        padding: 10px 15px;
    }
    /* .controls, li ve .icons ögeleri için padding değeri 0 piksel yükseklik, 2 piksel genişlik olarak ayarlanmıştır. */
    .controls,li,.icons{
        padding: 0px 2px;
    }
    /* .controls .icons ögesi için genişlik %20 olarak ayarlanmıştır. */
    .controls .icons{
        width: 20%;
    }
    /* .controls .icons i ögesi için genişlik 30 piksel, yazı tipi boyutu 12 piksel olarak ayarlanmıştır. */
    .controls .icons i{
        width: 30px;
        font-size: 12px;
    }
    /* .controls .row.to .icons ögesi için sağ kenardan 75 piksel boşluk bırakılır ve gap özelliği ile simgeler arasındaki boşluk ayarlanır. */
    .controls .row.to .icons{
        margin-right: 75px;
        gap: 5px;
    }
    /* .controls .row.from .icons ögesi için sol kenardan 5 piksel boşluk bırakılır ve gap özelliği ile simgeler arasındaki boşluk ayarlanır */
    .controls .row.from .icons{
        margin-left: 5px;
        gap: 5px;
    }
}
```



```

/* .controls .row.from select ögesi için yazı tipi boyutu 11 piksel
olarak ayarlanır ve sol kenardan 25 piksel geri kaydırılır. */
.controls .row.from select{
    font-size:11px;
    margin-left: -25px;
}
/* .controls .row.to select ögesi için yazı tipi boyutu 11 piksel
olarak ayarlanır ve sağ kenardan 20 piksel geri kaydırılır. */
.controls .row.to select{
    font-size:11px;
    margin-right: -20px;
}
/* .controls .exchange ögesi için yazı tipi boyutu 10 piksel olarak
ayarlanır. */
.controls .exchange{
    font-size: 10px;
}
/* .container button ögesi için genişlik 100%, padding değeri 8 piksel,
margin-top değeri 10 piksel ve yazı tipi boyutu 15 piksel olarak
ayarlanmıştır. */
.container button{
    width: 100%;
    padding: 8px;
    margin-top: 10px;
    font-size:15px;
}
}

```

JAVASCRIPT ÜLKE ADLAR

// Bu kod, her birinde bir dil kodu ve karşılık gelen dilin adı olan bir obje içerir. Bu kodlar, belirli bir dilin belirli bir yerde kullanıldığını tanımlamak için kullanılabilir.

```

const countries = {
    "am-ET": "Amharic",
    "ar-SA": "Arabic",
    "be-BY": "Biellarus",
    "bem-ZM": "Bemba",
    "bi-VU": "Bislama",
    "bjs-BB": "Bajan",
    "bn-IN": "Bengali",
    "bo-CN": "Tibetan",
    "br-FR": "Breton",
    "bs-BA": "Bosnian",

```

```
"ca-ES": "Catalan",
"cop-EG": "Coptic",
"cs-CZ": "Czech",
"cy-GB": "Welsh",
"da-DK": "Danish",
"dz-BT": "Dzongkha",
"de-DE": "German",
"dv-MV": "Maldivian",
"el-GR": "Greek",
"en-GB": "English",
"es-ES": "Spanish",
"et-EE": "Estonian",
"eu-ES": "Basque",
"fa-IR": "Persian",
"fi-FI": "Finnish",
"fn-FNG": "Fanagalo",
"fo-FO": "Faroese",
"fr-FR": "French",
"gl-ES": "Galician",
"gu-IN": "Gujarati",
"ha-NE": "Hausa",
"he-IL": "Hebrew",
"hi-IN": "Hindi",
"hr-HR": "Croatian",
"hu-HU": "Hungarian",
"id-ID": "Indonesian",
"is-IS": "Icelandic",
"it-IT": "Italian",
"ja-JP": "Japanese",
"kk-KZ": "Kazakh",
"km-KM": "Khmer",
"kn-IN": "Kannada",
"ko-KR": "Korean",
"ku-TR": "Kurdish",
"ky-KG": "Kyrgyz",
"la-VA": "Latin",
"lo-LA": "Lao",
"lv-LV": "Latvian",
"men-SL": "Mende",
"mg-MG": "Malagasy",
"mi-NZ": "Maori",
"ms-MY": "Malay",
"mt-MT": "Maltese",
"my-MM": "Burmese",
"ne-NP": "Nepali",
"niu-NU": "Niuean",
```

```
"nl-NL": "Dutch",  
"no-NO": "Norwegian",  
"ny-MW": "Nyanja",  
"ur-PK": "Pakistani",  
"pau-PW": "Palauan",  
"pa-IN": "Panjabi",  
"ps-PK": "Pashto",  
"pis-SB": "Pijin",  
"pl-PL": "Polish",  
"pt-PT": "Portuguese",  
"rn-BI": "Kirundi",  
"ro-RO": "Romanian",  
"ru-RU": "Russian",  
"sg-CF": "Sango",  
"si-LK": "Sinhala",  
"sk-SK": "Slovak",  
"sm-WS": "Samoan",  
"sn-ZW": "Shona",  
"so-SO": "Somali",  
"sq-AL": "Albanian",  
"sr-RS": "Serbian",  
"sv-SE": "Swedish",  
"sw-SZ": "Swahili",  
"ta-LK": "Tamil",  
"te-IN": "Telugu",  
"tet-TL": "Tetum",  
"tg-TJ": "Tajik",  
"th-TH": "Thai",  
"ti-TI": "Tigrinya",  
"tk-TM": "Turkmen",  
"tl-PH": "Tagalog",  
"tn-BW": "Tswana",  
"to-TO": "Tongan",  
"tr-TR": "Turkish",  
"uk-UA": "Ukrainian",  
"uz-UZ": "Uzbek",  
"vi-VN": "Vietnamese",  
"wo-SN": "Wolof",  
"xh-ZA": "Xhosa",  
"yi-YD": "Yiddish",  
"zu-ZA": "Zulu"
```

```
}
```

JAVASCRIPT KODLARI

```
// Bu kodlar, web sayfasındaki çeviri arayüzüne ilişkin öğelerin tanımlanması ve ülke seçeneklerinin dinamik olarak oluşturulması için kullanılıyor.
// const anahtar kelimesi ile değişkenler fromText, toText, exchangeIcon, selectTag, icons, translateBtn tanımlanıyor.
document.querySelector() fonksiyonuyla, HTML'deki ilgili öğeler seçiliyor ve değişkenlere atanıyor.
// document.querySelectorAll() fonksiyonuyla, sayfadaki tüm select etiketleri ve .row i sınıfına sahip olan tüm ikonlar selectTag ve icons adlı değişkenlere atanıyor.
const fromText = document.querySelector(".from-text"),
toText = document.querySelector(".to-text"),
exchangeIcon = document.querySelector(".exchange"),
selectTag = document.querySelectorAll("select"),
icons = document.querySelectorAll(".row i");
translateBtn = document.querySelector("button"),
// forEach() döngüsü kullanılarak selectTag dizisindeki her bir select etiketi için bir dizi ülke seçeneği oluşturuluyor. for...in döngüsü kullanarak, countries nesnesindeki her bir öge üzerinde geziniliyor ve her bir seçenek, option değişkenine atanıyor. insertAdjacentHTML() fonksiyonuyla, option değişkeni, tag değişkeninin sonuna ekleniyor.
selectTag.forEach((tag, id) => {
  for (let country_code in countries) {
    let selected = id == 0 ? country_code == "en-GB" ? "selected" : "" : country_code == "tr-TR" ? "selected" : "";
    let option = `<option ${selected}
value="${country_code}">${countries[country_code]}</option>`;
    tag.insertAdjacentHTML("beforeend", option);
  }
});
// Bu kod bloğu, "exchange" adlı bir simgeye tıklanıldığında, iki metin kutusunun (fromText ve toText) içeriğinin yerlerini değiştiren bir işlevi tanımlar. Ayrıca, iki dil seçeneği arasında da bir değiş tokuş gerçekleştirir.
exchangeIcon.addEventListener("click", () => {
  let tempText = fromText.value,
  tempLang = selectTag[0].value;
  fromText.value = toText.value;
  toText.value = tempText;
  selectTag[0].value = selectTag[1].value;
  selectTag[1].value = tempLang;
});
// Bu kod bloğu, fromText adlı input alanına herhangi bir tuşa basıldığında tetiklenen bir olay dinleyicisi ekler. Eğer fromText
```

değeri boşsa (!fromText.value), toText değeri de boşaltılır (toText.value = ""), yani çeviri sonucu da boş olacaktır. Bu kod bloğu, çeviri işleminin doğru şekilde yapılabilmesi için gereklidir.

```
fromText.addEventListener("keyup", () => {
  if(!fromText.value) {
    toText.value = "";
  }
});
// Bu kod, translateBtn ögesine bir tıklama olayı dinleyicisi ekler.
// Düğmeye tıklandığında işlev yürütülür.
translateBtn.addEventListener("click", () => {
  let text = fromText.value.trim(),
  translateFrom = selectTag[0].value,
  translateTo = selectTag[1].value;
  if(!text) return;
  toText.setAttribute("placeholder", "Translating...");
  let apiUrl =
`https://api.mymemory.translated.net/get?q=${text}&langpair=${translate
From}|${translateTo}`;
  fetch(apiUrl).then(res => res.json()).then(data => {
    toText.value = data.responseData.translatedText;
    data.matches.forEach(data => {
      if(data.id === 0) {
        toText.value = data.translation;
      }
    });
    toText.setAttribute("placeholder", "Translation");
  });
});
//Bu kod JavaScript dilinde yazılmış bir event listener fonksiyonudur.
//Kodlar bir butona (translateBtn) bir click eventi ekler. Butona
//tıklandığında, fromText input elementinin değeri alınır ve trim()
//fonksiyonu kullanılarak başındaki ve sonundaki boşluklar temizlenir.
//Ayrıca, selectTag adında bir dizi oluşturulur ve ilk elemanı,
//translateFrom olarak atanan selectTag[0].value ile eşleştirilirken,
//ikinci elemanı translateTo olarak atanan selectTag[1].value ile
//eşleştirilir.
icons.forEach(icon => {
  icon.addEventListener("click", ({target}) => {
    if(!fromText.value || !toText.value) return;
    if(target.classList.contains("fa-copy")) {
      if(target.id == "from") {
        navigator.clipboard.writeText(fromText.value);
      } else {
        navigator.clipboard.writeText(toText.value);
      }
    }
  });
});
```

```
    } else {
        let utterance;
        if(target.id == "from") {
            utterance = new
SpeechSynthesisUtterance(fromText.value);
            utterance.lang = selectTag[0].value;
        } else {
            utterance = new SpeechSynthesisUtterance(toText.value);
            utterance.lang = selectTag[1].value;
        }
        speechSynthesis.speak(utterance);
    }
});
});
```