

[Sign up](#)[Code](#)[Issues](#) 0[Pull requests](#) 0[Projects](#) 0[Actions](#)[Wiki](#)[Security](#)[Pulse](#)

Branch: master ▾

[Find file](#)[Copy path](#)[cpp\\_kursu\\_odevleri](#) / [templates](#) / templates\_08.md

Fetching contributors...



31 lines (23 sloc) | 834 Bytes

[Raw](#)[Blame](#)[History](#)

**Şablon tür parametresi (*T*) olan türün *Nec* isimli bir içsel türe (*nested type*) sahip olup olmadığını sorgulayacak *hasTypeNec* isimli sınıf şablonunu kodlayınız:**

```
template <typename T>
struct hasTypeNec {
    //...
    static constexpr bool value = expr;
};
```

**Sınıfın *constexpr* olan *bool* türden *value* isimli ögesi, eğer şablon tür parametresi olan *T* türü *Nec* isimli bir içsel türe sahip ise *true* aksi halde *false* değere sahip olmalı. Yazdığınız sınıf şablonunu aşağıdaki test kodu ile sınavınız:**

```
struct A {
    typedef int Nec;
};

struct B {};
```

```
#include <iostream>

int main()
{
    constexpr bool b1 = hasTypeNec<A>::value;
    constexpr bool b2 = hasTypeNec<B>::value;
    constexpr bool b3 = hasTypeNec<int>::value;

    std::cout << b1 << b2 << b3; //100
}
```

[Privacy](#)  
[Security](#)  
[Status](#)  
[Help](#)

[Contact GitHub](#)  
[Pricing](#)  
[API](#)  
[Training](#)  
[Blog](#)  
[About](#)