Tntnet users guide, ccccc s

Table of contents

Concept

```
name;
street;
city = “New York”;
int age;      // content of text-input is converted to int
int sport[];  // multiple checkboxes with the same name on my form
</%args>
<# use the vector like this: #>
% for (sport_type::const_iterator it = sport.begin(); it !=
sport.end(); ++it) {
<$ *it $>
% }
```

Components

Ecpp-pages are called **components**. They are identified by their name. The name is composed of the class-name and the library-name divided with '@'.

Components, which are called by Tntnet are called top-level-components.

Top-level-components return the http-response-code. If a explicit return-statement is not specified the constant HTTP_OK is returned. Constants, which define http-response-codes are defined in the class

Basically there are several possibilities to call a component.

In the simplest case the component-name

```
}>
<& greeting name=(nameValue) lastname=(lastnameValue)
repeat=(repeatNum)>

<%def greeting>
<%args>
name;
lastname;
name;
```

```
std::string currentUser;
</%session>
```

Request-variables

```

```

```

```

```

```

variable tntnet answers every requexp×@ 404 – HTTP_NOT_FOUND.MapUrl” takes at leaxp×@

pa

passed to a outputstream and you