

How should I inject?

Constructor injection

```
public function __construct(IMailer $mailer)
```

Method injection (inject* method)

```
public function injectMailer(IMailer $mailer)
```

Property injection (@inject annotation)

```
/** @var \Nette\Mail\IMailer @inject */  
public $mailer;
```

What should I use?	Presenter	Service	Component with factory
Constructor	★★★★ ¹⁾	★★★★	★★★★
Inject method	★★★ ³⁾	★ ²⁾	★ ^{2) 3)}
@inject	★★★ ³⁾	✖	✖

1) only in the final presenters, not in the BasePresenter

2) it is required to enable this using `inject: true` in service definition in your neon file

3) only for the BasePresenter or the BaseControl, Prefer constructor injection elsewhere

★★★ - recommended

★ - avoid this

★★ - you can use this

✖ - do not use

What should I inject?

(declared class/interface inject according to the left column)

Cache (doc)

Require `Nette\Caching\IStorage`

```
//$storage contains IStorage  
$cache = new Nette\Caching\Cache($storage);  
$cache->load($key);
```

HTTP (doc)

HTTP Request `Nette\Http\IRequest`

- `getUrl()` - `Nette\Http\UrlScript`
- `getHeaders()` - array of headers
- `getHeader($header)`
- `getCookie($key, $default = NULL)`
- `getRemoteAddress()`
- ...

HTTP Response `Nette\Http\IResponse`

- `setCode($code)`
- `setCookie($cookie, $value, $expire, ...)`
- `setHeader($header, $value)`
- `isSent()`
- ...

Mailing (doc)

Require `Nette\Mail\IMailer`

```
//$mailer contains IMailer  
$message = new Nette\Mail\Message();  
$message->addTo($to);  
...  
$mailer->send($message);
```

Session (doc)

Require `Nette\Http\Session`

```
//$session contains Session  
$section = $session->getSection($mySection);  
$section->key = $value;
```

Database (doc)

Require `Nette\Database\Context`

- `query($query[, $arg[, ...]])` - executes query
- `table($name)` - returns `Nette\Database\Selection`

User and security (doc)

User `Nette\Security\User`

- `isLoggedIn()`
- `getIdentity()`
- `getRoles()`
- `isInRole($role)`
- `isAllowed($resource, $privilege)`
- `login($id = NULL, $password = NULL)`
- `logout($clearIdentity = FALSE)`

Authorizator `Nette\Security\IAuthorizator`

Authenticator `Nette\Security\IAuthenticator`

User storage `Nette\Security\IUserStorage`

Application

Application `Nette\Application\Application`

- `getPresenter()` - returns current presenter if any

Router `Nette\Application\IRouter`

- `constructUrl($appRequest, $refUrl)` - creates URL for given application request

LinkGenerator

`Nette\Application\LinkGenerator`

- `link($dest, $params)` - generates url for the presenter

Templates

`Nette\Application\UI\ITemplateFactory`

- `createTemplate($control = NULL)` - creates **template** you know from the presenter

`Nette\Bridges\ApplicationLatte\ILatteFactory`

- `create()` - creates `Latte\Engine` instance

Nette DI cheatsheet (2.3)

© 2015 [David Matějka](#)

License [CC BY-SA 3.0](#)