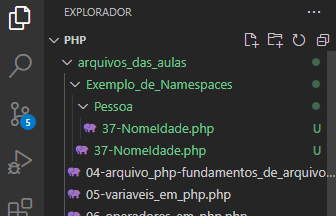
Antes veja como está a organização dos arquivos nas pastas...



EXEMPLO 01

<?php

//NAMESPACE:

//O conceito de namespace não é um conceito único do PHP, na verdade namespace é um conceito geral para organização e utilização de pastas de arquivos. Com um namespace podemos determinar que um arquivo só será reconhecido á partir de um nome especial que damos no início do caminho dele.

//Como o namespace é usado?

//Quando temos muitos arquivos armazenados em pastas dentro da nossa aplicação e ainda por cima utilizamos bibliotecas externas em conjunto com os nossos arquivos a chance de ter nomes de arquivos idênticos pode ser grande. Por exemplo, imagine que temos um arquivo para uma classe "CPF" e além disso estamos usando uma biblioteca externa para validar "CPF", a chance de existir 2 arquivos "CPF" é grande.

//Agora imagine que dentro dos nossos arquivos conseguíssemos identificar a qual pasta eles pertencem e qual caminho o PHP tem que seguir para encontrar um determinado arquivo, bom, não precisa imaginar, esse método existe e é por através dos namespaces.

    require\_once "Exemplo\_de\_Namespaces/Pessoa/37-NomeIdade.php";

    //Note que o caminho com require\_once está referenciando para um arquivo que está dentro da pasta "Pessoa"

    use MinhaOrganizacaoPessoal\arquivo\_das\_aulas\Exemplos\_de\_Namespaces\NomeIdade;

    //Note que ele só vai aceitar o require por que usamos o caminho correto de namespace, se por exemplo tirássemos o "MinhaOrganizacaoPessaol", o php não iria mais importar o arquivo...

    //Se você reparar bem, quando desejamos usar um elemento específico de um arquivo importado junto ao namespace, temos que usar como último nome do caminho o nome do elemento que queremos importar - que no nosso caso aqui é uma classe. Note que o nome dessa classe é idêntico ao nome de outra classe que também está dentro da pasta "Exemplos\_de\_Namespaces", por isso o php irá comparar se o caminho informado no require e no namespace combinam para que a classe correta seja chamada...

    $carlos = new NomeIdade('Carlos', '42');

    echo $carlos->getInformacoes();

?>

ARQUIVO EXPORTADO...

<?php

    namespace MinhaOrganizacaoPessoal\arquivo\_das\_aulas\Exemplos\_de\_Namespaces;

    //Note aqui não utilizamos o "MinhaOrganizacaoPessoal", essa pasta não existe, nós que inventamos esse caminho pessoal para que o namespace correto seja informado ao php para referenciar esse arquivo e só depois ele começa a busca nas pastas...

    //Outro detalhe importante, os namespaces sempre são acessados por através de contra-barra, não importa o Sistema Operacional

    class NomeIdade {

        private $nome;

        private $idade;

        public function \_\_construct(string $nome, string $idade){

            $this->nome = $nome;

            $this->idade = $idade;

        }

        public function getInformacoes():string

        {

            return "Essa Pessoa se chama: $this->nome têm $this->idade ano(s) de idade." . PHP\_EOL;

        }

    }

?>

[Running] php "c:\Users\Almoxarifado\Documents\php\arquivos\_das\_aulas\37-namespaces\_exemplo\_01.php"

Essa Pessoa se chama: Carlos têm 42 ano(s) de idade.

[Done] exited with code=0 in 0.092 seconds

EXEMPLO 02

<?php

    require\_once "Exemplo\_de\_Namespaces/37-NomeIdade.php";

    //Note que o caminho no require mudou, agora ele não entra na pasta "Pessoa"...

    use MinhaOrganizacaoPessoal\arquivo\_das\_aulas\Exemplos\_de\_Namespaces\NomeIdade;

    //Mas usamos o mesmo namespace que vai entrar na pasta "Exemplos\_de\_Namespaces" á procura por um elemento "NomeIdade", o primeiro que ela encontrar será o que vai trazer para nós...

    $carlos = new NomeIdade('Carlos', '42');

    echo $carlos->getInformacoes();

    //Veja que Carlos não é mais uma pessoa, ele é um animal agora...

?>

ARQUIVO EXPORTADO...

<?php

    namespace MinhaOrganizacaoPessoal\arquivo\_das\_aulas\Exemplos\_de\_Namespaces;

    //Note que temos outro arquivo com o mesmo nome de Classe e dentro da mesma pasta "Exemplos\_de\_Namespaces"...

    class NomeIdade {

        private $nome;

        private $idade;

        public function \_\_construct(string $nome, string $idade){

            $this->nome = $nome;

            $this->idade = $idade;

        }

        public function getInformacoes():string

        {

            return "Esse Animal se chama: $this->nome têm $this->idade ano(s) de idade." . PHP\_EOL;

            //Mas se referenciada corretamente ela traz um resultado totalmente diferente...

        }

    }

?>

[Running] php "c:\Users\Almoxarifado\Documents\php\arquivos\_das\_aulas\37-namespaces\_exemplo\_02.php"

Esse Animal se chama: Carlos têm 42 ano(s) de idade.

[Done] exited with code=0 in 0.089 seconds