(a) PHP if else

In PHP, the `if-else` statement allows you to execute certain code blocks based on conditions.

```php
$age = 20;

if ($age > = 18) {
    echo "You are an adult.";
} else {
    echo "You are a minor.";
}
```

(b) PHP elseif

`elseif` is a conditional statement used in PHP to check multiple conditions in succession. It's essentially a combination of `else` and `if`, allowing you to check another condition if the preceding `if` statement evaluates to false

```php
$score = 75;

if ($score > = 90) {
    echo "A";
} elseif ($score > = 80) {
    echo "B";
} elseif ($score > = 70) {
    echo "C";
} elseif ($score > = 60) {
    echo "D";
} else {
```

```php
        echo "F";
    }
```

(c) PHP elseif vs else if

In PHP, `elseif` and `else if` are equivalent and can be used interchangeably. Both are used to combine conditions in an if-else statement. The choice between them is purely a matter of personal preference or coding style.

(d) Nested if else

Nested if-else statements are a way to include one or more if-else statements inside another if or else block. This allows for more complex conditional logic by evaluating multiple conditions within different branches of the code.

```php
$score = 75;
$grade = "";

if ($score > = 90) {
    $grade = "A";
} else {
    if ($score  > = 80) {
        $grade = "B";
    } else {
        if ($score  > = 70) {
            $grade = "C";
        } else {
            if ($score  > = 60) {
                $grade = "D";
            } else {
                $grade = "F";
            }
```

```php
            }
        }
    }

    echo "Your grade is: $grade";
```

#php