

Date is : 13:32:20

CONVERT HTML TO PDF

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{(?:\s*"([^"])*"\s*:\s*"([^"])*"/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*\{.*\s*"([^"])*"\s*:\s*\{.*\s*"([^"])*"/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*\{\s*"([^"])*"\s*:\s*"([^"])*"/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*\{[^"]*\s*$/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1
```



```

monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.column });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{[^"]*"[^"]*\s*:\s*"[^"]"/);
    return createDependencyProposals();
  }
});

```

```

monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.column });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{[^"]*"[^"]*\s*:\s*"[^"]"/);
    return createDependencyProposals();
  }
});

```

```

monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.column });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{[^"]*"[^"]*\s*:\s*"[^"]"/);
    return createDependencyProposals();
  }
});

```

```

monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.column });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{[^"]*"[^"]*\s*:\s*"[^"]"/);
    return createDependencyProposals();
  }
});

```

```

monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.column });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*{[^"]*"[^"]*\s*:\s*"[^"]"/);
    return createDependencyProposals();
  }
});

```

$\}) ;$

```
monaco.languages.registerCompletionItemProvider('json', {
    provideCompletionItems: function (model, position) {
        // find out if we are completing a property in the 'dependencies' object.
        var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
        var match = textUntilPosition.match(/"dependencies"\s*:\s*\{[^}]*\}/);
        return createDependencyProposals();
    }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacterIndex() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*\{[^}]*\s*"([^"]*)"(\s*:\s*[^,]*\s*)?/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```

```
monaco.languages.registerCompletionItemProvider('json', {
  provideCompletionItems: function (model, position) {
    // find out if we are completing a property in the 'dependencies' object.
    var textUntilPosition = model.getValueInRange({ startLineNumber: 1, startColumn: 1, endLineNumber: 1, endColumn: position.getCharacter() });
    var match = textUntilPosition.match(/"dependencies"\s*:\s*\{\s*"([^"])*"\s*:\s*"([^"])*"/);
    if (match) {
      return createDependencyProposals();
    }
    return [];
  }
});
```