```
Needs["Quantum`Notation`"];

SetQuantumAliases[];
```

Quantum Notation Palette

```
buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
   {"Needs[\"Quantum`Notation`\"]",
    "SpanFromLeft",
    "SetQuantumAliases[]",
    "SpanFromLeft",
    "SetQuantumObject[□]",
    "SpanFromLeft",
    "SetQuantumScalar[□]",
    "SpanFromLeft",

    "QuantumPartialTrace[□,□̂]",
    "SpanFromLeft",
    "QuantumPartialTranspose[□,□̂]",
    "SpanFromLeft",

    "QuantumReplace[□,{ | □〉:→ | □〉, | □〉:→ | □〉}]",
    "SpanFromLeft",
    "SpanFromLeft",
    "SpanFromLeft",

    "DefineOperatorOnKets[□,{ | □〉:→ | □〉, | □〉:→ | □〉}]",
    "SpanFromLeft",
    "SpanFromLeft",
    "SpanFromLeft",

    " | □〉", "〈□ |", " | □_□̂〉", "〈□_□̂ |",
    " | □_□̂, □_□̂〉", "〈□_□̂, □_□̂ |", " | □_□̂, □_□̂, □_□̂〉", "〈□_□̂, □_□̂, □_□̂ |",
    "〈□ | □〉", " | □〉·〈□ |", "〈□_□̂ |· | □_□̂〉", " | □_□̂〉·〈□_□̂ |",
    "〈□_□̂, □_□̂ |· | □_□̂, □_□̂〉", " | □_□̂, □_□̂〉·〈□_□̂, □_□̂ |", " | □_□̂〉· | □_□̂〉", "〈□_□̂ |·〈□_□̂ |",
    "□̂· | □_□̂〉", "□_□̂", "□̂", "Tr_□̂[□]",
    "∑_□^□ □", "⊗_□^□ □", "‖□‖", "□/‖□‖",
    "(□)†", "(□)*", "⟦□, □⟧_", "⟦□, □⟧+",
    "·", "⊗", "□·□", "□⊗□"
   }] /. HoldPattern[Button["SpanFromLeft", ___]] :→ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 4]], WindowTitle → "Quantum Notation"]

NotebookObject[ Quantum Notation ]
```

Quantum Algebra Palette

```
buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Notation`\"]",
     "SpanFromLeft",
     "SetQuantumObject[□]",
     "SetQuantumObject[□,□,□]",
     "DefineOperatorOnKets[□,{ | □⟩⧴ | □⟩, | □⟩⧴ | □⟩}]",
     "SpanFromLeft",
     "CollectFromLeft[□]",
     "CollectFromRight[□]",
     "Expand[□]",
     "CommutatorExpand[□]",
     "CommutatorExpand[□,Anticommutators→True]",
     "SpanFromLeft",
     "CommutatorExpand[□,ReverseOrdering→True]",
     "SpanFromLeft",
     "CommutatorExpand[□,NestedCommutators→True]",
     "SpanFromLeft",
     "EvaluateCommutators[□]",
     "EvaluateAllCommutators[□]",
     "FactorKet[□]",
     "CollectKet[□]",
     "FactorKetList[□]",
     "Simplify[□]",
     "⟦□, □⟧_-",
     "⟦□, □⟧_+",
     "⟦□, □⟧_-=□",
     "⟦□, □⟧_+=□",
     "□·□=□",
     "□^□=□",
     "·",
     "⊗"
    }] /. HoldPattern[Button["SpanFromLeft", ___]] ⧴ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]], WindowTitle → "Quantum Algebra"]
```

NotebookObject[Quantum Algebra]

Quantum to Matrix Palette

buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,

{"Needs[\"Quantum`Notation`\"]",

"SpanFromLeft",

"MatrixToDirac[$\begin{pmatrix} \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \end{pmatrix}$ **,{4}]",**

"MatrixToDirac[$\begin{pmatrix} \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \end{pmatrix}$ **,{2,2}]",**

"TensorToDirac[$\begin{pmatrix} \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \end{pmatrix}$ **]",**

"TensorToDirac[$\begin{pmatrix} \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} & \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} \\ \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} & \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} \end{pmatrix}$ **]",**

"VectorToDirac[{□,□,□,□},{4}]",

"VectorToDirac[{□,□,□,□},{2,2}]",

"MatrixToDirac[$\begin{pmatrix} \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \end{pmatrix}$ **,{2,2},** $\{0_{\hat{1}}\to a1_{\hat{a}}, 1_{\hat{1}}\to a2_{\hat{a}}, 0_{\hat{2}}\to b1_{\hat{b}}, 1_{\hat{2}}\to b2_{\hat{b}}\}$ **]",**

"SpanFromLeft",

"TensorToDirac[$\begin{pmatrix} \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} & \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} \\ \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} & \begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix} \end{pmatrix}$ **,** $\{0_{\hat{1}}\to a1_{\hat{a}}, 1_{\hat{1}}\to a2_{\hat{a}}, 0_{\hat{2}}\to b1_{\hat{b}}, 1_{\hat{2}}\to b2_{\hat{b}}\}$ **]",**

"SpanFromLeft",

"MatrixToDirac[□,{□}]",
"TensorToDirac[□]",

"DiracToMatrix[□, $\{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]",**
"DiracToTensor[□, $\{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]",**

"MatrixForm[DiracToMatrix[□, $\{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]]",**
"MatrixForm[DiracToTensor[□, $\{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]]",**

"DiracEigensystem[□, $\{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]",**
"DiracToVector[$\square \mid \square_{\hat{\square}}, \square_{\hat{\square}}\rangle + \square \mid \square_{\hat{\square}}, \square_{\hat{\square}}\rangle, \{\{\square_{\hat{\square}},\square_{\hat{\square}}\},\{\square_{\hat{\square}},\square_{\hat{\square}}\}\}$ **]"**

}] /. HoldPattern[Button["SpanFromLeft", ___]] :→ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]], WindowTitle → "Quantum to Matrix"]

```
NotebookObject[ Quantum to Matrix ]
```

Quantum Tests and Measurements

```
Needs["Quantum`Notation`"];

buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Notation`\"]",
     "SpanFromLeft",
     "QuantumScalarQ[□]",
     "KetQ[□]",
     "QuantumMeasurement[□ | □_□̂, □_□̂, □_□̂⟩+□ | □_□̂, □_□̂, □_□̂⟩,{□̂,□̂}]",
     "SpanFromLeft",
     "QuantumMeasurement[□,{□},Assumptions→And[□≠□,□≠□]]",
     "SpanFromLeft",
     "QuantumMeasurement[□,{□},FactorKet→False]",
     "SpanFromLeft",
     "QuantumDensityOperator[QuantumMeasurement[□,{□}]]",
     "SpanFromLeft",
     "Part[QuantumMeasurement[□,{□}],1]",
     "SpanFromLeft"
    }] /. HoldPattern[Button["SpanFromLeft", ___]] :→ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]],
 WindowTitle → "Quantum Tests and Measurements"]

NotebookObject[ Quantum Tests and Measurements ]
```

Quantum Computing Commands

```
Needs["Quantum`Computing`"];

Names["Quantum`Computing`*"]
```

```
buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Computing`\"]",
     "SetComputingAliases[]",
     "QuantumEvaluate[□]",
     "QuantumTable[□]",
     "PauliExpand[□]",
     "QuantumTableForm[□]",
     "FactorKet[□]",
     "TraditionalForm[QuantumTableForm[□]]",
     "QuantumMatrixForm[□]",
     "QuantumTensorForm[□]",
     "QuantumMatrix[□]",
     "QuantumTensor[□]",
     "MatrixQuantum[□]",
     "TensorQuantum[□]",
     "QubitToDec[ | □_□̂, □_□̂, □_□̂⟩]",
     "DecToQubit[□,□]",
     "QuantumPartialTrace[□,□̂]",
     "QuantumPartialTranspose[□,□̂]",
     "QuantumPlot[□]",
     "SetQuantumGate[□,□]",
     "QuantumPlot3D[□]",
     "SetQuantumGate[□,{□,□}]",
     "SetQuantumGate[□,□,Function[{□,□},□]]",
     "SpanFromLeft",
     "QuantumPlot[QubitMeasurement[□,{□̂,□̂}]]",
     "SpanFromLeft",
     "QuantumEvaluate[QubitMeasurement[□,{□̂,□̂}]]",
     "SpanFromLeft",
     "QuantumEvaluate[QubitMeasurement[□,{□̂,□̂},FactorKet→False]]",
     "SpanFromLeft",
     "QuantumDensityOperator[QubitMeasurement[□,{□̂,□̂}]]",
     "SpanFromLeft"
    }] /. HoldPattern[Button["SpanFromLeft", ___]] :→ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]],
 WindowTitle → "Quantum Computing Commands"]
```

```
NotebookObject[ Quantum Computing Commands ]
```

```
QubitToDec[ | 1_1̂, 0_2̂, 1_3̂⟩]
```

```
5
```

```
DecToQubit[9, 5]
```

```
| 0_1̂, 1_2̂, 0_3̂, 0_4̂, 1_5̂⟩
```

```
Needs["Quantum`Computing`"]
```

```
Welcome to Quantum`Computing`
A Mathematica package for Quantum Computing
  in Dirac bra-ket notation and plotting of quantum circuits
by José Luis Gómez-Muñoz

Execute SetComputingAliases[] in order to use
  the keyboard to enter quantum objects in Dirac's notation
SetComputingAliases[] must be executed again in each new notebook that is created
```

TraditionalForm$\left[\textbf{QuantumTableForm}\left[\mathcal{H}_{\hat{1}} \otimes \mathcal{H}_{\hat{2}}\right]\right]$

	Input	Output
0	$\lvert 00\rangle$	$\frac{1}{2}\lvert 00\rangle + \frac{1}{2}\lvert 01\rangle + \frac{1}{2}\lvert 10\rangle + \frac{1}{2}\lvert 11\rangle$
1	$\lvert 01\rangle$	$\frac{1}{2}\lvert 00\rangle - \frac{1}{2}\lvert 01\rangle + \frac{1}{2}\lvert 10\rangle - \frac{1}{2}\lvert 11\rangle$
2	$\lvert 10\rangle$	$\frac{1}{2}\lvert 00\rangle + \frac{1}{2}\lvert 01\rangle - \frac{1}{2}\lvert 10\rangle - \frac{1}{2}\lvert 11\rangle$
3	$\lvert 11\rangle$	$\frac{1}{2}\lvert 00\rangle - \frac{1}{2}\lvert 01\rangle - \frac{1}{2}\lvert 10\rangle + \frac{1}{2}\lvert 11\rangle$

QuantumTable$\left[\mathcal{H}_{\hat{1}} \otimes \mathcal{H}_{\hat{2}}\right]$

$$\Big\{\Big\{\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\Big\},$$
$$\Big\{\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\Big\},$$
$$\Big\{\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\Big\},$$
$$\Big\{\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\Big\}\Big\}$$

QuantumTableForm$\left[\mathcal{H}_{\hat{1}} \otimes \mathcal{H}_{\hat{2}}\right]$

	Input	Output
0	$\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
1	$\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
2	$\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
3	$\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$

FactorKet$\left[\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\right]$

$$\text{Hold}\left[\left(\lvert 0_{\hat{1}}\rangle + \lvert 1_{\hat{1}}\rangle\right) \otimes \left(\frac{1}{2}\left(\lvert 0_{\hat{2}}\rangle + \lvert 1_{\hat{2}}\rangle\right)\right)\right]$$

$$\left(\lvert 0_{\hat{1}}\rangle + \lvert 1_{\hat{1}}\rangle\right) \otimes \left(\frac{1}{2}\left(\lvert 0_{\hat{2}}\rangle + \lvert 1_{\hat{2}}\rangle\right)\right)$$

$$\frac{1}{2}\left(\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle\right) + \frac{1}{2}\left(\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\right)$$

QuantumEvaluate$\left[\textbf{QubitMeasurement}\left[\mathcal{H}_{\{\hat{1},\hat{2}\}}, \{\hat{2}\}\right]\right]$

QuantumMeasurement::nonket: The first argument

$$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 0_{\hat{2}}\rvert + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 0_{\hat{2}}\rvert + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 0_{\hat{2}}\rvert + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 0_{\hat{2}}\rvert + \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle\cdot$$
$$\langle 0_{\hat{1}}, 1_{\hat{2}}\rvert - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 1_{\hat{2}}\rvert + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle\cdot\langle 0_{\hat{1}}, 1_{\hat{2}}\rvert - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 0_{\ll 1\gg}, 1_{\ll 1\gg}\rvert + \frac{1}{2}\ll 1\gg\cdot\ll 1\gg + \frac{1}{2}\lvert 0_{\hat{1}}$$

is not a linear combination of kets. $, 1_{\hat{2}}\rangle\cdot\langle 1_{\hat{1}}, 0_{\hat{2}}\rvert - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle\cdot\langle 1_{\hat{1}}, 0_{\hat{2}}\rvert - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 1_{\hat{1}}, 0_{\hat{2}}\rvert + \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle\cdot\langle 1_{\hat{1}}, 1_{\hat{2}}\rvert - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle\cdot\langle 1_{\hat{1}}, 1_{\hat{2}}\rvert - \frac{1}{2}\rvert$

```
$Aborted
```

```
QuantumEvaluate[QubitMeasurement[ℋ_{1̂,2̂} · |0_1̂, 0_2̂⟩, {2̂}]]
```

Probability	Measurement	State
$\frac{1}{2}$	$\{\{0_{\hat{2}}\}\}$	$(\,\lvert 0_{\hat{1}}\rangle + \lvert 1_{\hat{1}}\rangle) \otimes \frac{\lvert 0_{\hat{2}}\rangle}{\sqrt{2}}$
$\frac{1}{2}$	$\{\{1_{\hat{2}}\}\}$	$(\,\lvert 0_{\hat{1}}\rangle + \lvert 1_{\hat{1}}\rangle) \otimes \frac{\lvert 1_{\hat{2}}\rangle}{\sqrt{2}}$
Probability	Measurement	State

```
QuantumEvaluate[QubitMeasurement[ℋ_{1̂,2̂} · |0_1̂, 0_2̂⟩, {2̂}, FactorKet → False]]
```

Probability	Measurement	State
$\frac{1}{2}$	$\{\{0_{\hat{2}}\}\}$	$\frac{\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle}{\sqrt{2}}$
$\frac{1}{2}$	$\{\{1_{\hat{2}}\}\}$	$\frac{\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle}{\sqrt{2}}$
Probability	Measurement	State

```
? FactorKet
```

FactorKet[expr] factors tensor products of kets in expr. Its output is returned inside a Hold command

```
? SetQuantumGate
```

After evaluating SetQuantumGate[symbol,narg] symbol will be treated as
quantum gate of narg arguments (qubits) by QuantumEvaluate[] and other functions
SetQuantumGate[symbol,narg,Function[{q1,q2...},diracexpr]] replaces symbol with diracexpr
(evaluated in q1,q2...) when symbol is part of the argument of QuantumEvaluate.
SetQuantumGate[symbol,{n1,n2}] and SetQuantumGate[symbol,{n1,n2},Function[{q1,q2...},diracexpr]]
define symbol as a quantum gate with a number of arguments n1<=narg<=n2

```
QuantumTable[ℋ_1̂ · ℋ_2̂]
```

$$\{\{\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\},$$
$$\{\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\},$$
$$\{\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\},$$
$$\{\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle, \frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle\}\}$$

```
QuantumTableForm[ℋ_1̂ · ℋ_2̂]
```

	Input	Output
0	$\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
1	$\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
2	$\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$
3	$\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$	$\frac{1}{2}\lvert 0_{\hat{1}}, 0_{\hat{2}}\rangle - \frac{1}{2}\lvert 0_{\hat{1}}, 1_{\hat{2}}\rangle - \frac{1}{2}\lvert 1_{\hat{1}}, 0_{\hat{2}}\rangle + \frac{1}{2}\lvert 1_{\hat{1}}, 1_{\hat{2}}\rangle$

Quantum Computing Gates

```
Needs["Quantum`Computing`"];
```

SetComputingAliases[]

```
ALIASES:
[ESC]on[ESC]        Quantum concatenation
  symbol (operator application, inner product and outer product)
[ESC]qket0[ESC]     Ket of qubit 0 template
[ESC]qbra0[ESC]     Bra of qubit 0 template
[ESC]qket1[ESC]     Ket of qubit 1 template
[ESC]qbra1[ESC]     Bra of qubit 1 template
[ESC]qket[ESC]      Ket of qubit template
[ESC]qqket[ESC]     Ket of two qubits template
[ESC]qqqket[ESC]    Ket of three qubits template
[ESC]qbra[ESC]      Bra of qubit template
[ESC]qqbra[ESC]     Bra of two qubits template
[ESC]qqqbra[ESC]    Bra of three qubits template
[ESC]toqb[ESC]      Base-10 Integer to binary qubit template
[ESC]ket[ESC]       Ket template
[ESC]bra[ESC]       Bra template
[ESC]qb[ESC]        Qubit template
[ESC]qv[ESC]        Qubit-value template
[ESC]qketbra[ESC]   Element of a one-qubit operator template
[ESC]qqketbra[ESC]  Element of a two-qubits operator template
[ESC]qqqketbra[ESC] Element of a three-qubits operator template
[ESC]k+[ESC]        Plus ket (eigenstate of the first Pauli matrix)
[ESC]b+[ESC]        Plus bra
[ESC]k-[ESC]        Minus ket (eigenstate of the first Pauli matrix)
[ESC]b-[ESC]        Minus bra
[ESC]k00[ESC]       Ket of Bell State 00
[ESC]k01[ESC]       Ket of Bell State 01
[ESC]k10[ESC]       Ket of Bell State 10
[ESC]k11[ESC]       Ket of Bell State 11
[ESC]b00[ESC]       Bra of Bell State 00
[ESC]b01[ESC]       Bra of Bell State 01
[ESC]b10[ESC]       Bra of Bell State 10
[ESC]b11[ESC]       Bra of Bell State 11
[ESC]kphi+[ESC]     Ket of Bell State phi+
[ESC]kpsi+[ESC]     Ket of Bell State psi+
[ESC]kphi-[ESC]     Ket of Bell State phi-
[ESC]kpsi-[ESC]     Ket of Bell State psi-
[ESC]bphi+[ESC]     Bra of Bell State phi+
[ESC]bpsi+[ESC]     Bra of Bell State psi+
[ESC]bphi-[ESC]     Bra of Bell State phi-
[ESC]bpsi-[ESC]     Bra of Bell State psi-
[ESC]her[ESC]       Hermitian conjugate template
[ESC]con[ESC]       Complex conjugate template
[ESC]norm[ESC]      Quantum norm template
[ESC]trace[ESC]     Partial trace template
[ESC]tp[ESC]        Tensor-product symbol
[ESC]tprod[ESC]     Tensor-product template
```

```
[ESC]tprodqb[ESC]   Tensor-product of Qubit template
[ESC]tpow[ESC]      Tensor-power template
[ESC]tpowqb[ESC]    Tensor-power of Qubit template
[ESC]s0[ESC]        0th-Pauli operator (Identity) template
[ESC]s1[ESC]        1st-Pauli operator (X) template
[ESC]s2[ESC]        2nd-Pauli operator (Y) template
[ESC]s3[ESC]        3rd-Pauli operator (Z) template
[ESC]so[ESC]        0th-Pauli operator (Identity) template
[ESC]sx[ESC]        1st-Pauli operator (X) template
[ESC]sy[ESC]        2nd-Pauli operator (Y) template
[ESC]sz[ESC]        3rd-Pauli operator (Z) template
[ESC]sp[ESC]        General Pauli operator template
[ESC]ig[ESC]        Identity gate template
[ESC]xg[ESC]        Pauli-X gate
[ESC]yg[ESC]        Pauli-Y gate
[ESC]zg[ESC]        Pauli-Z gate
[ESC]hg[ESC]        Haddamard gate
[ESC]pg[ESC]        Parametric phase gate
[ESC]sg[ESC]        S Phase gate
[ESC]tg[ESC]        T π/8 gate
[ESC]swap[ESC]      Swap gate
[ESC]cgate[ESC]     Controlled-Gate template
[ESC]ccgate[ESC]    Controlled-controlled-Gate template
[ESC]cccgate[ESC]   Controlled-controlled-controlled-Gate template
[ESC]cnot[ESC]      Controlled-Not template
[ESC]ccnot[ESC]     Controlled-controlled-Not template
[ESC]cccnot[ESC]    Controlled-controlled-controlled-Not template
[ESC]toff[ESC]      Toffoli gate
[ESC]fred[ESC]      Fredkin gate
[ESC]qg[ESC]        Quantum gate of one argument
[ESC]qqg[ESC]       Quantum gate of one argument applied to two qubits
[ESC]qqqg[ESC]      Quantum gate of one argument applied to three qubits
[ESC]qgg[ESC]       Quantum gate of two arguments
[ESC]qggg[ESC]      Quantum gate of three arguments
[ESC]pqg[ESC]       Parametric quantum gate of one argument
[ESC]qr[ESC]        Quantum register template
[ESC]qrg[ESC]       Quantum-register gate template

SetComputingAliases[] must be executed again in
  each new notebook that is created, only one time per notebook.
```

Names["Quantum`Computing`*"]

```
{DecToQubit, MatrixQuantum, PauliExpand, PauliIdentities, QuantumBackground,
 QuantumConnectionStyle, QuantumControlStyle, QuantumEigensystem,
 QuantumEigensystemForm, QuantumGatePowers, QuantumGateShifting, QuantumGateStyle,
 QuantumMatrix, QuantumMatrixForm, QuantumMeter, QuantumMeterStyle, QuantumNotStyle,
 QuantumPlot, QuantumPlot3D, QuantumSparseArray, QuantumSwapStyle, QuantumTable,
 QuantumTableForm, QuantumTensor, QuantumTensorForm, QuantumTextStyle,
 QuantumVerticalTextStyle, QuantumWireStyle, QubitLabels, QubitList, QubitMeasurement,
 QubitToDec, SetComputingAliases, SetQuantumGate, TensorQuantum, zz020CeroOneQ,
 zz020Controlled, zz020MultiQubit, zz020TensorPower, zz020TP, zz020TPdat, zz020TPend,
 zz020TPini, zz020TwoScalarsListQ, zz050Bell, zz050Superpos, C, FREDKIN, H, I,
 NOT, P, QFT, Register, s, S, SWAP, T, TOFFOLI, X, Y, Z, σ, Φ, Ψ, 11, 10, 0, 01, 00}
```

```
buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Computing`\"]",
     "SpanFromLeft",
     "SpanFromLeft",
     "SpanFromLeft",

     "ℐ_□̂",
     "𝒳_□̂",
     "𝒴_□̂",
     "𝒵_□̂",

     "σ_{0,□̂}",
     "σ_{𝒳,□̂}",
     "σ_{𝒴,□̂}",
     "σ_{𝒵,□̂}",

     "ℋ_□̂",
     "𝒮_□̂",
     "𝒯_□̂",
     "𝒮𝒲𝒜𝒫_{□̂,□̂}",

     "𝒬ℱ𝒯_□̂",
     "𝒬ℱ𝒯_{□̂,□̂}",
     "𝒬ℱ𝒯_{□̂,□̂,□̂}",
     "𝒬ℱ𝒯_{□̂,□̂,□̂,□̂}",

     "□_□̂",
     "□_{□̂,□̂}",
     "□_{□̂,□̂,□̂}",
     "□_{□̂,□̂,□̂,□̂}",

     "□_{{□̂,□̂}}",
     "□_{{□̂,□̂,□̂}}",
     "𝒞^{{□̂}}[□]",
     "𝒞^{{□̂,□̂}}[□]",

     "𝒞^{{□̂}}[𝒩𝒪𝒯_□̂]",
     "𝒯𝒪ℱℱ𝒪ℒℐ_{□̂,□̂,□̂}",
     "ℱℛℰ𝒟𝒦ℐ𝒩_{□̂,□̂,□̂}",
     "𝒫_□̂[□]",

     "·",
     "⊗",
     "ℛegister[□,□,□,□]",
     "(□)^†"

    }] /. HoldPattern[Button["SpanFromLeft", ___]] ⧴ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 4]], WindowTitle → "Quantum Computing Gates"]

NotebookObject[ Quantum Computing Gates ]
```

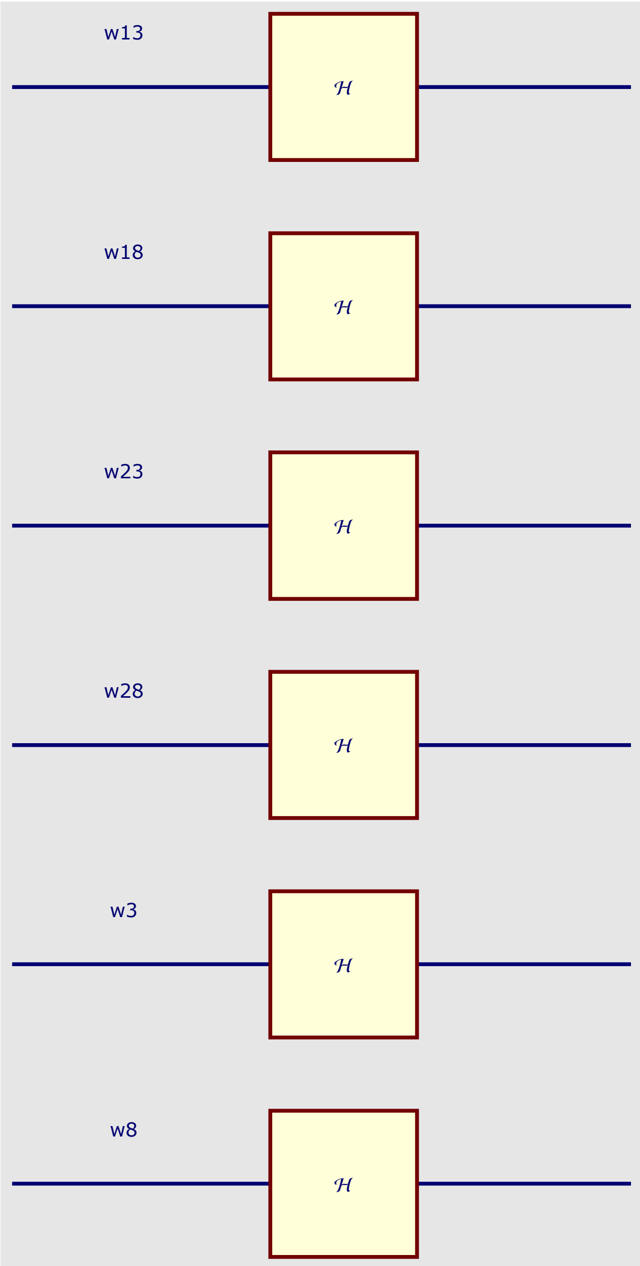
```
QuantumEvaluate[Register[3, 30, 5, w]]
```

$\{\hat{w3}, \hat{w8}, \hat{w13}, \hat{w18}, \hat{w23}, \hat{w28}\}$

$\mathcal{H}_{Register[\hat{3},30,5,w]}$

$\mathcal{H}_{Register[3,30,5,w]}$

```
QuantumPlot[%]
```



Quantum Computing Kets

```
Needs["Quantum`Computing`"];
```

```
buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
   {"Needs[\"Quantum`Computing`\"]",
    "SpanFromLeft",
    "SpanFromLeft",
    "SpanFromLeft",

    " | 0□̂〉",
    " | 1□̂〉",
    "〈0□̂ |",
    "〈1□̂ |",

    " | +□̂〉",
    " | -□̂〉",
    "〈+□̂ |",
    "〈-□̂ |",

    " | Φ+□̂,□̂〉",
    " | Ψ+□̂,□̂〉",
    " | Φ-□̂,□̂〉",
    " | Ψ-□̂,□̂〉",

    "〈Φ+□̂,□̂ |",
    "〈Ψ+□̂,□̂ |",
    "〈Φ-□̂,□̂ |",
    "〈Ψ-□̂,□̂ |",

    " | ℬ00,□̂,□̂〉",
    " | ℬ01,□̂,□̂〉",
    " | ℬ10,□̂,□̂〉",
    " | ℬ11,□̂,□̂〉",

    "〈ℬ00,□̂,□̂ |",
    "〈ℬ01,□̂,□̂ |",
    "〈ℬ10,□̂,□̂ |",
    "〈ℬ11,□̂,□̂ |",

    "⨂_{n=□}^{□} | □n̂〉",
    "⨂_{□=□}^{□} □",
    "( | □□̂〉)^⊗□",
```

```
      "(□)⊗□",

      " | □⟩□",
      "⟨□ |□",
      " | □⟩{□̂,□̂}",
      " | □⟩{□̂,□̂,□̂}",

      " | □□̂⟩",
      " | □□̂, □□̂⟩",
      " | □□̂, □□̂, □□̂⟩",
      " | □⟩",

      "⟨□□̂ |",
      "⟨□□̂, □□̂ |",
      "⟨□□̂, □□̂, □□̂ |",
      "⟨□ |",

      " | □□̂⟩·⟨□□̂ |",
      " | □□̂, □□̂⟩·⟨□□̂, □□̂ |",
      " | □□̂, □□̂, □□̂⟩·⟨□□̂, □□̂, □□̂ |",
      "SpanFromLeft",

      "Tr□̂[□]",
      "‖□‖",
      "(□)†",
      "(□)*",

      "·",
      "⊗",
      "□□̂",
      "□̂"

    }] /. HoldPattern[Button["SpanFromLeft", ___]] ⧴ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 4]], WindowTitle → "Quantum Computing Kets"]
```

Machote de paleta Quantum Computing

```
Needs["Quantum`Computing`"];

buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Computing`\"]",
     "SpanFromLeft"
    }] /. HoldPattern[Button["SpanFromLeft", ___]] ⧴ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]], WindowTitle → "Quantum Computing"]
```

Machote de Paleta Notation

```
Needs["Quantum`Notation`"];

buttonList = Map[Button[#, NotebookWrite[InputNotebook[], #], Appearance → "Palette"] &,
    {"Needs[\"Quantum`Notation`\"]",
     "SpanFromLeft"
    }] /. HoldPattern[Button["SpanFromLeft", ___]] ⧴ SpanFromLeft;
CreatePalette[Grid[Partition[buttonList, 2]], WindowTitle → "Quantum"]
```