Test Bed for Quat Package

Regression Tests

In[1]:=
```
Get[ "Quat.m", Path -> { NotebookDirectory [] } ];
```

Binary operator "⟷": assert equality between arguments as

In[2]:=
```
SetAttributes[ LongLeftRightArrow, HoldAllComplete ];
```

In[3]:=
```
LongLeftRightArrow[ a_, b_ ] := Module[
   { result },
   If[ a == b, Return[] ];
   result = {
      { "Left:", HoldForm[a] },
      { "→", Style[ a, FontColor → Red ] },
      { "", "≠" },
      { "Right:", HoldForm[b] },
      { "→", Style[ b, FontColor → Red ] }
    } // TableForm;
   CellPrint[ ExpressionCell[ result, "Output" ] ];
   FrontEndExecute[ FrontEndToken[ "EvaluatorAbort" ] ]
  ];
```

Sanity-checks

Operate on a quaternion constructed from an axis vector **u** and an angle θ

In[4]:=
```
q = ToQ$AngleAxis[ θ, { x, y, z } ];
```

In[5]:=
```
1 ⟷ Simplify[ |q|,
  Assumptions → -π ≤ θ ≤ π ⋀ x² + y² + z² == 1 ⋀ { x, y, z, θ } ∈ Reals
 ]
```

In[6]:=
```
Abs[Sin[θ/2]] ⟷ FullSimplify[ Norm[ q⃗ ],
  Assumptions → -π ≤ θ ≤ π ⋀ x² + y² + z² == 1 ⋀ { x, y, z, θ } ∈ Reals
 ]
```

In[7]:=
```
q ⟷ Q[Cos[θ/2], x Sin[θ/2], y Sin[θ/2], z Sin[θ/2]]
```

In[8]:=
```
|q| ⟷ Sqrt[Cos[θ/2]^2 + x^2 Sin[θ/2]^2 + y^2 Sin[θ/2]^2 + z^2 Sin[θ/2]^2]
```

In[9]:=
```
Norm[q] ⟷ Sqrt[Cos[θ/2]^2 + x^2 Sin[θ/2]^2 + y^2 Sin[θ/2]^2 + z^2 Sin[θ/2]^2]
```

In[10]:=
```
Re[q] ⟷ Cos[θ/2]
```

In[11]:=
```
〈q〉 ⟷ Cos[θ/2]
```

In[12]:=
```
Im[q] ⟷ {x Sin[θ/2], y Sin[θ/2], z Sin[θ/2]}
```

In[13]:=

$$\vec{q} \leftrightarrow \left\{x\,\mathrm{Sin}\left[\frac{\theta}{2}\right],\ y\,\mathrm{Sin}\left[\frac{\theta}{2}\right],\ z\,\mathrm{Sin}\left[\frac{\theta}{2}\right]\right\}$$

In[14]:=

$$\mathrm{Abs\$Im}[q] \leftrightarrow \sqrt{x^2\,\mathrm{Sin}\left[\frac{\theta}{2}\right]^2 + y^2\,\mathrm{Sin}\left[\frac{\theta}{2}\right]^2 + z^2\,\mathrm{Sin}\left[\frac{\theta}{2}\right]^2}$$

In[15]:=

```
{re, im} = {
   Re[Complex[1, 1]^1.5],
   Im[Complex[1, 1]^1.5]
  };
Q[ 1, 1, 0, 0 ]^1.5 ↔ Q[ re, im, 0, 0 ]
Q[ 1, 0, 1, 0 ]^1.5 ↔ Q[ re, 0, im, 0 ]
Q[ 1, 0, 0, 1 ]^1.5 ↔ Q[ re, 0, 0, im ]
Clear[re, im]
```

In[20]:=

```
( Q[1, 2, 3, 4] ** 1 ** Q[1, 2, 3, 4] ) ↔
 ( Q[1, 2, 3, 4] ** Q[1, 0, 0, 0] ** Q[1, 2, 3, 4] )
```

In[21]:=

```
( Q[1, 2, 3, 4] ** {1, 2, 3} ** Q[1, 2, 3, 4] ) ↔
 ( Q[1, 2, 3, 4] ** {0, 1, 2, 3} ** Q[1, 2, 3, 4] )
```

In[22]:=

$$\left(Q[1, 2, 3, 4] ** \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \end{pmatrix} ** Q[1, 2, 3, 4] \right) \leftrightarrow$$

```
 ( Q[1, 2, 3, 4] ** Q[ 0, 1, 2, 3] ** Q[1, 2, 3, 4] )
```

In[23]:=

```
q =.;
```

▼ Orientation derivative according "Notes..."

If q is multiplied by the angular velocity ω from left side $\frac{1}{2}\,\omega\,q$, it means that ω components are in the inertial frame of reference

In[24]:=

```
Block[
 {
  ω = Q[ ωw, ωx, ωy, ωz ],
  q = Q[ w, x, y, z ]
 },
```

$$\frac{1}{2}\,\omega ** q == \mathrm{ToQ}\left[\frac{1}{2}\begin{pmatrix} w & -x & -y & -z \\ x & w & z & -y \\ y & -z & w & x \\ z & y & -x & w \end{pmatrix} . \begin{pmatrix} \omega w \\ \omega x \\ \omega y \\ \omega z \end{pmatrix}\right] \mathbin{//} \mathrm{Assert}$$

```
]
```

Out[24]:=

⊦ True ∎

In[25]:=

```
Q[ w, x, y, z ] // QForm
```

Out[25]:=

$(w \mid x\ \ y\ \ z)$

In[26]:=

```
ToMatrix@Q[ w, x, y, z ] // QForm
```

Out[26]:=

$$\left(\begin{array}{c|ccc} w & x & y & z \\ \hline -x & w & -z & y \\ -y & z & w & -x \\ -z & -y & x & w \end{array}\right)$$