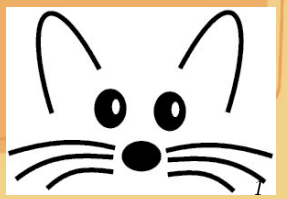
平成17年｜一日大学院

スクイークでコンピュータを操ろう!!

知識科学教育研究センター
金井｜秀明



授業の流れ

- コンピュータを操るには？
- スクイークって何？
- プロジェクト
  - 車を作ろう
  - 車を運転しよう
  - 利口な車を作ろう
- まとめ

コンピュータを操るには？

- ソフトウェア（ソフト）
  - 市販のソフト（ワープロ，ゲームなど）を使う
  - 自分で作ったソフトを使う．

「自分でソフトを作る」
＝「コンピュータプログラミング：プログラミング」

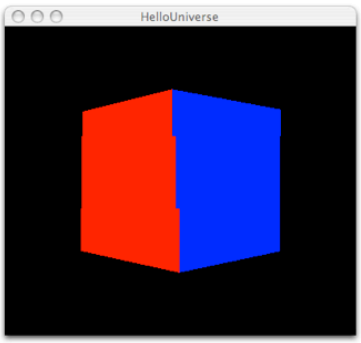
プログラミングは？

- 「コンピュータをどのように動かしたいか」（命令）を書く．
  - 命令は，プログラミング言語（BAISC,¦C言語，Java言語…）で書きます．

例：プログラミング

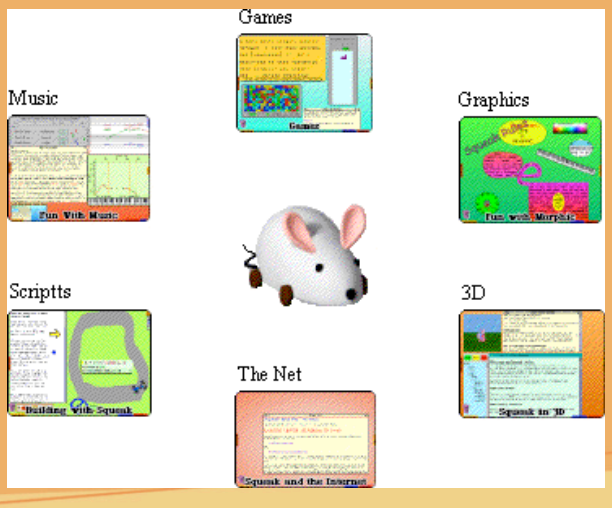
HelloUniverse.java

```
TransformGroup objTrans = new TransformGroup();
objTrans.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
objRoot.addChild(objTrans);

// Create a simple Shape3D node; add it to the scene graph.
objTrans.addChild(new ColorCube(0.4));

// Create a new Behavior object that will perform the
// desired operation on the specified transform and add
// it into the scene graph.
Transform3D yAxis = new Transform3D();
Alpha rotationAlpha = new Alpha(-1, 4000);

RotationInterpolator rotator =
    new RotationInterpolator(rotationAlpha, objTrans, yAxis,
                0.0f, (float) Math.PI*2.0f);
BoundingSphere bounds =
    new BoundingSphere(new Point3d(0.0,0.0,0.0), 100.0);
rotator.setSchedulingBounds(bounds);
objRoot.addChild(rotator);

    // Have Java 3D perform optimizations on this scene graph.
    objRoot.compile();

return objRoot;
}

public HelloUniverse() {
}

public void init() {
setLayout(new BorderLayout());
    GraphicsConfiguration config =
       SimpleUniverse.getPreferredConfiguration();

Canvas3D c = new Canvas3D(config);
add("Center", c);

// Create a simple scene and attach it to the virtual universe
BranchGroup scene = createSceneGraph();
u = new SimpleUniverse(c);

    // This will move the ViewPlatform back a bit so the
    // objects in the scene can be viewed.
    u.getViewingPlatform().setNominalViewingTransform();

u.addBranchGraph(scene);
}
```

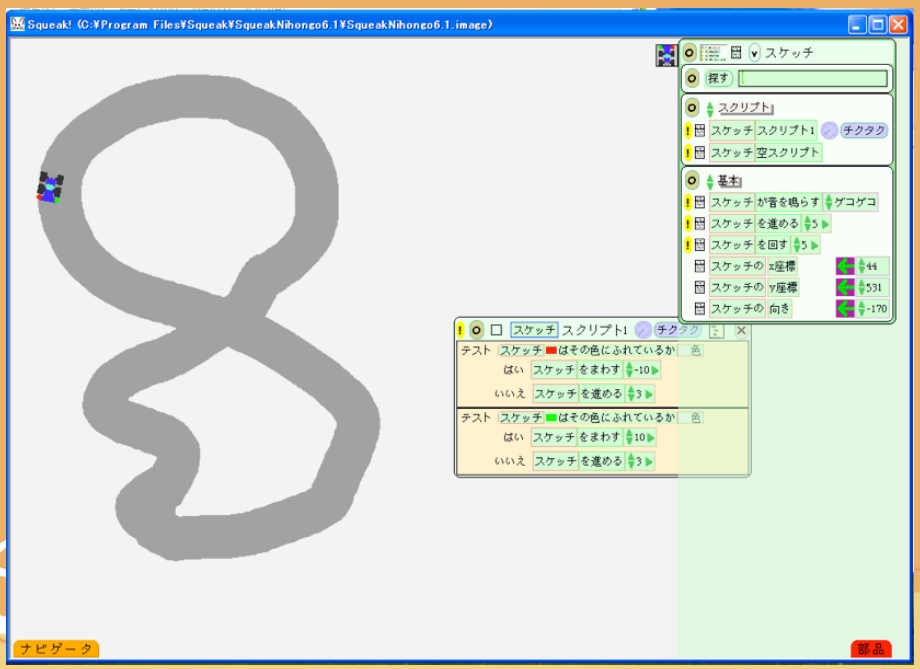


スクイークって何？

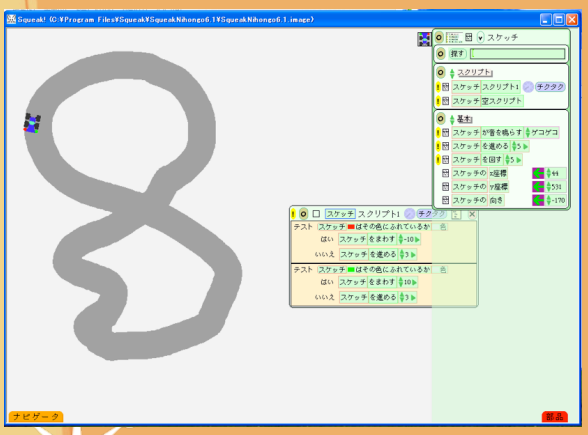
- プログラミング言語です．
- 例

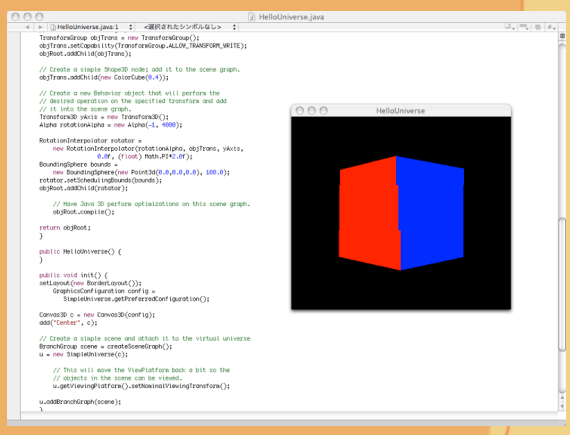


スクイークの例

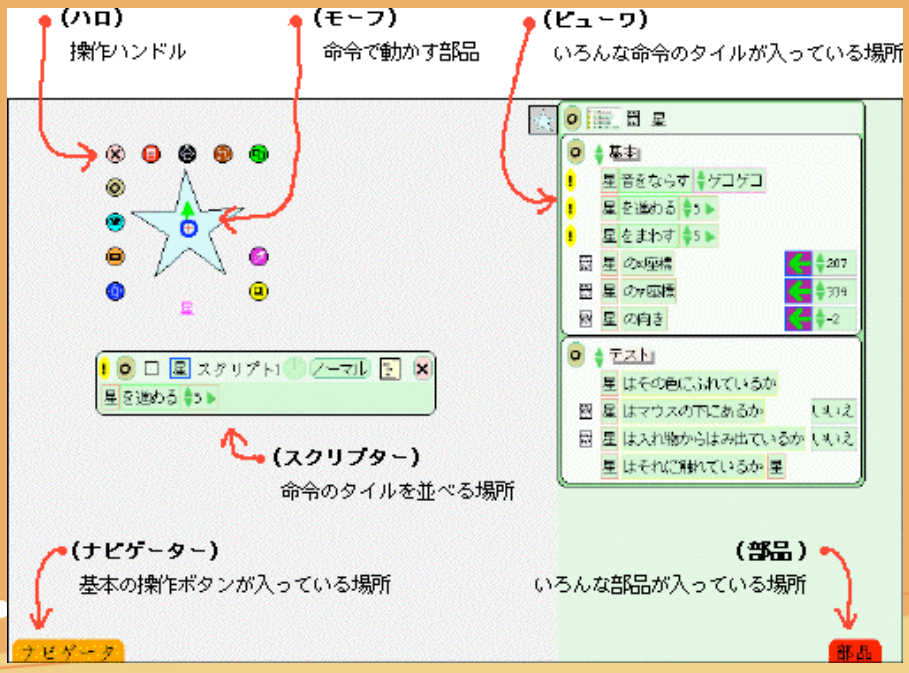


スクイークと
他のプログラム言語





スクイークの画面



今日の授業では，

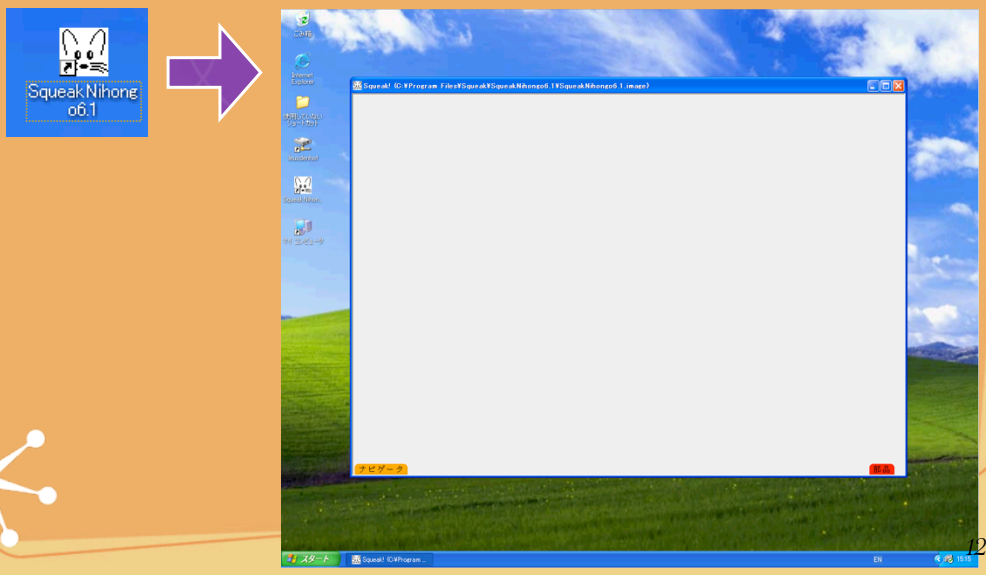
- 「スクイーク」を使って，プログラミングを勉強します．
  - ほとんど文字（命令）は書きません．
  - 絵を描いて，その絵がどう動くかを指定するだけ．
  - コンピュータの中で，車を動かすことができます．

プロジェクト

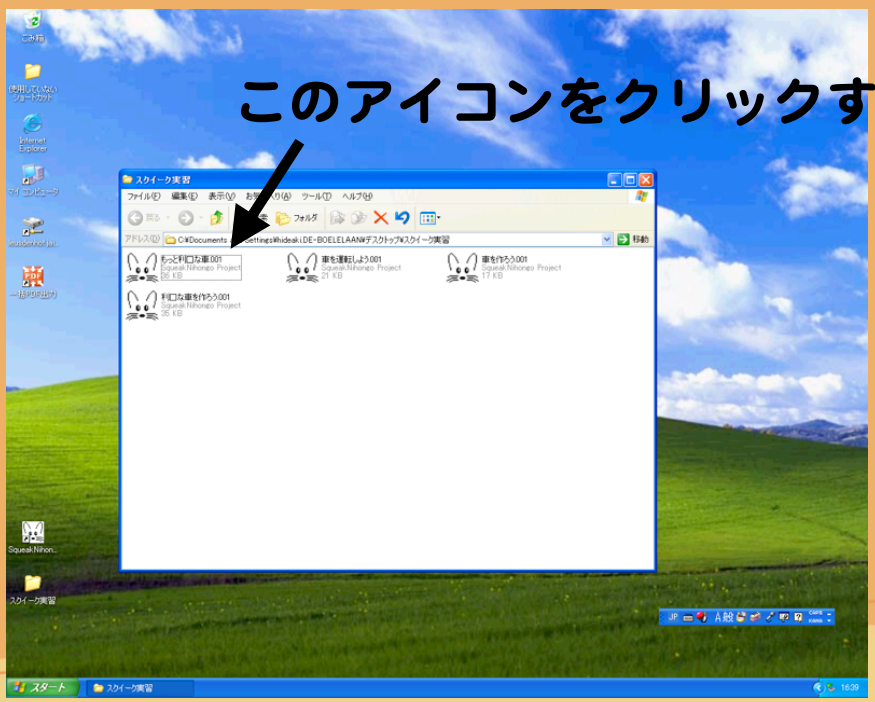
- 車を作ろう
- 車を運転しよう
- 利口な車を作ろう

準備：スクイークの起動方法

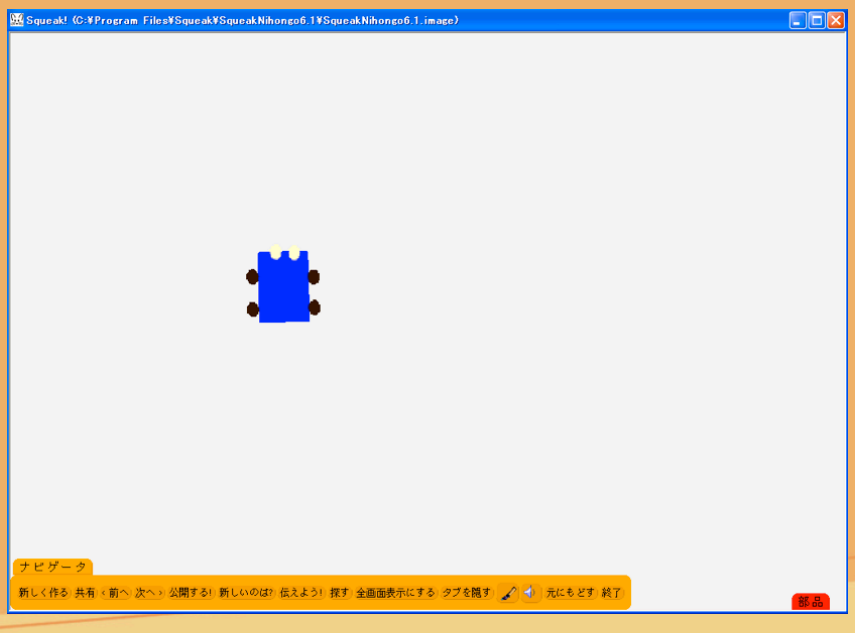
- Squeakのアイコンをクリックする.



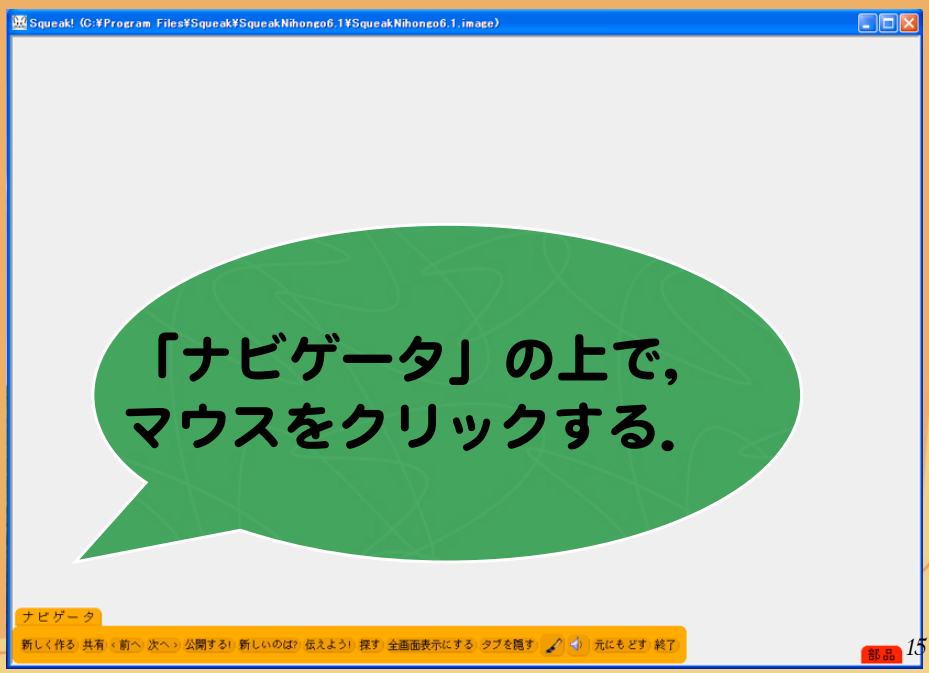
実習例


このアイコンをクリックする

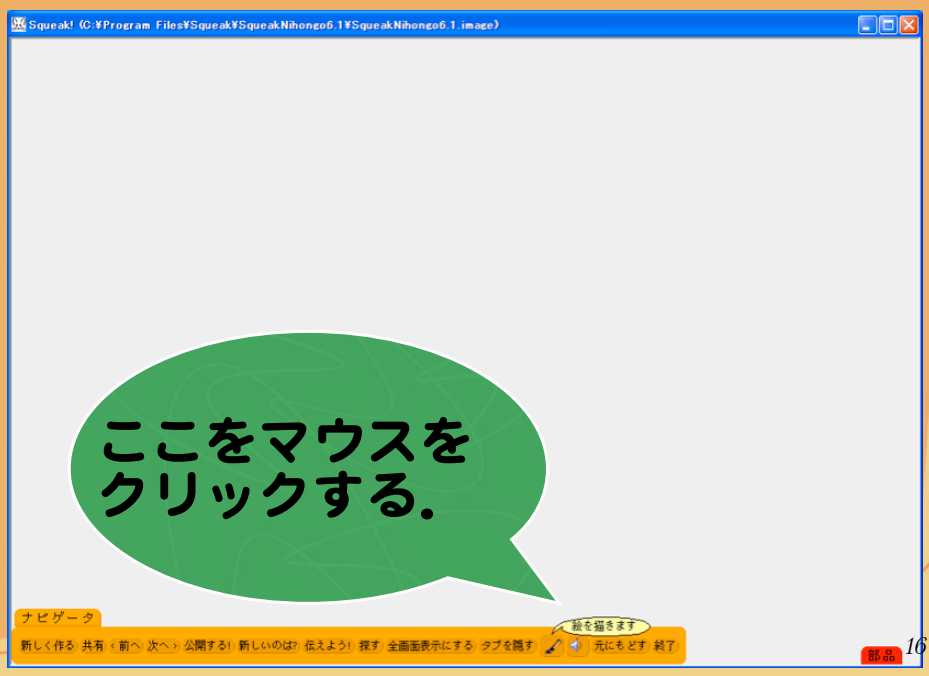
車を作ろう


Squeak! (C:¥Program Files¥Squeak¥SqueakNihongo6.1¥SqueakNihongo6.1.image)
ナビゲータ
新しく作る 共有 前へ 次へ 公開する! 新しいのは? 伝えよう! 探す 全画面表示にする タブを隠す 元にもどす 終了
部品

お絵描きツールの起動１

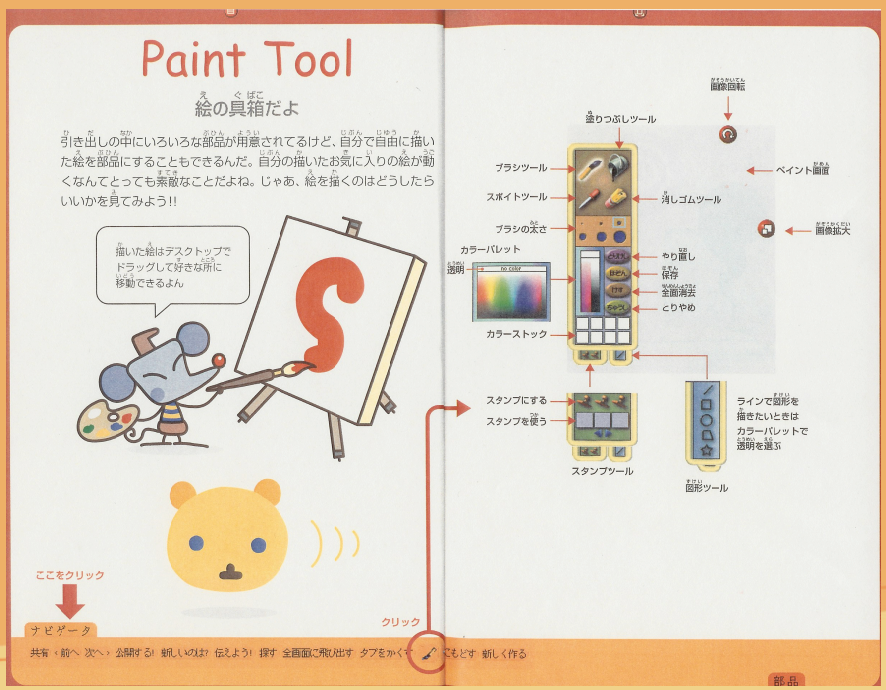


お絵描きツールの起動2

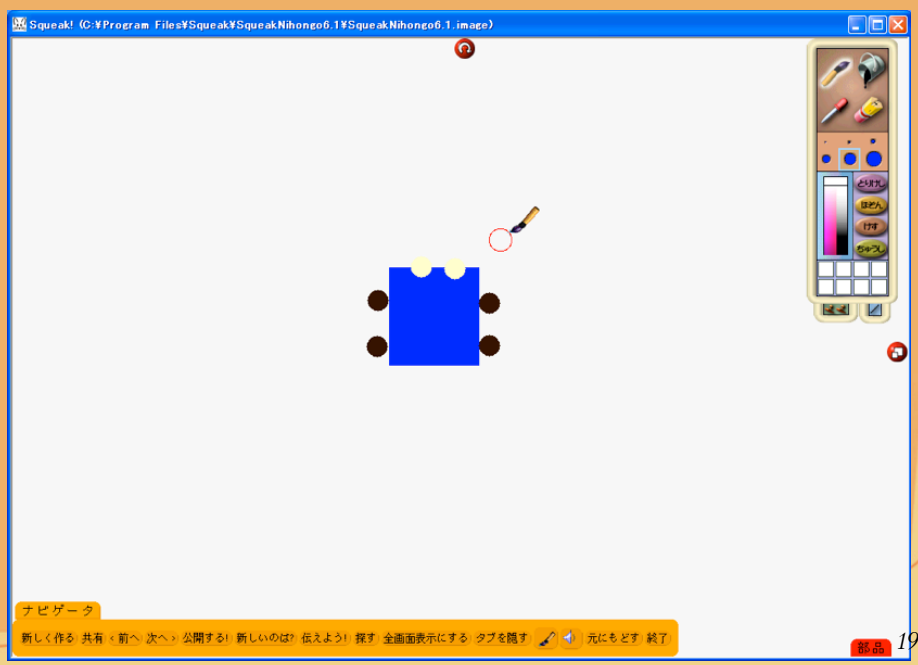


お絵描きツールの起動3

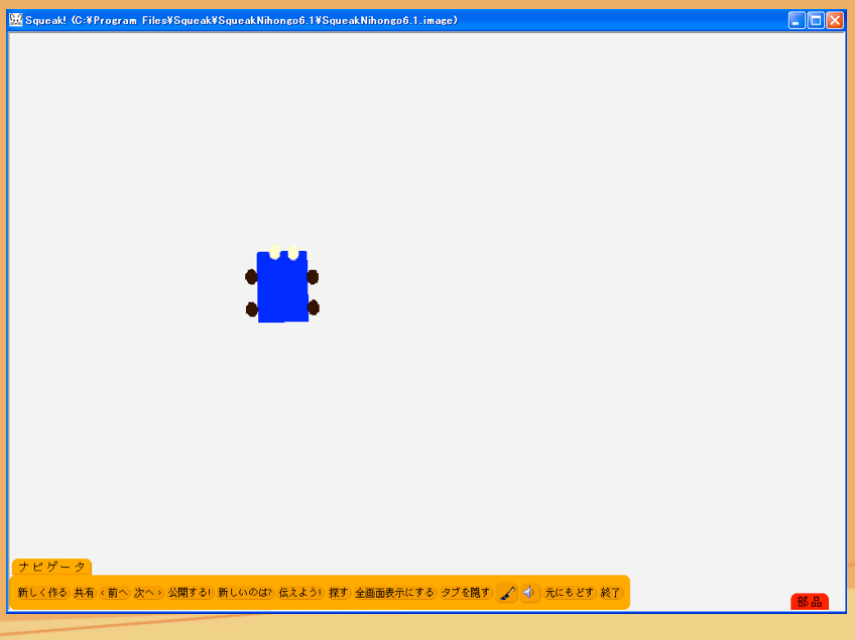
お絵描きツールの説明



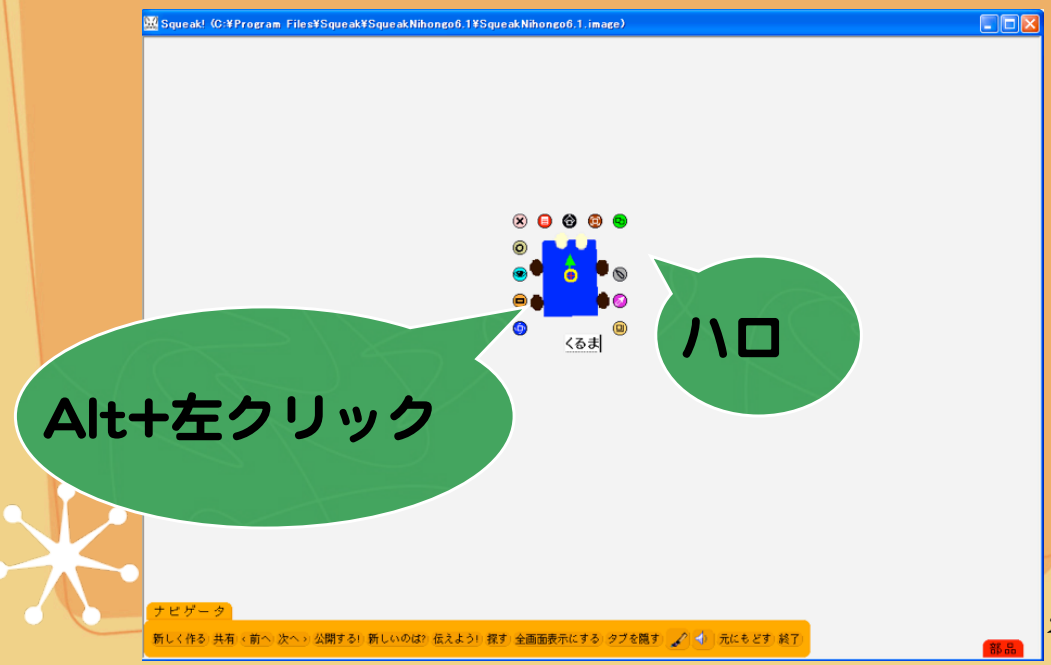
お絵描きツール



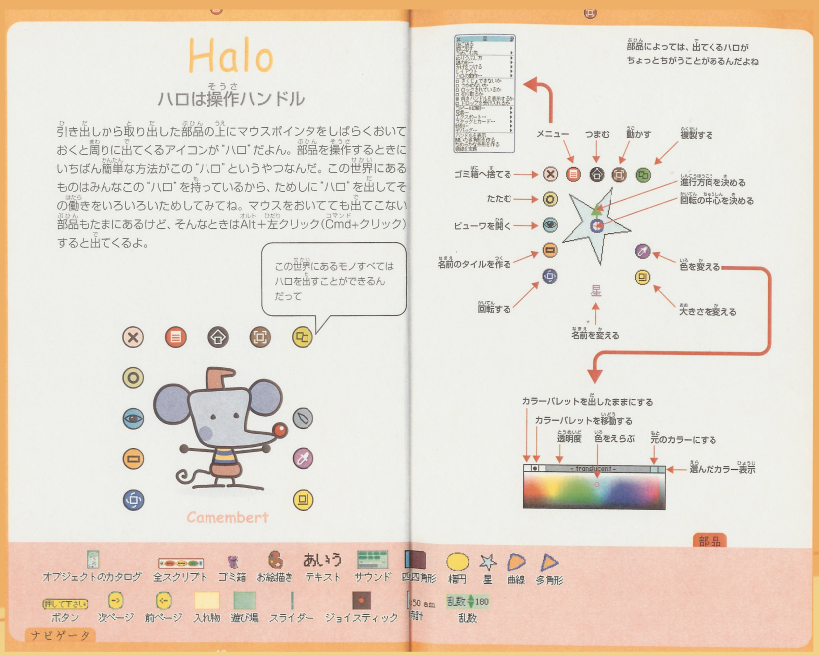
車ができました。



コントロールツール “ハロ”



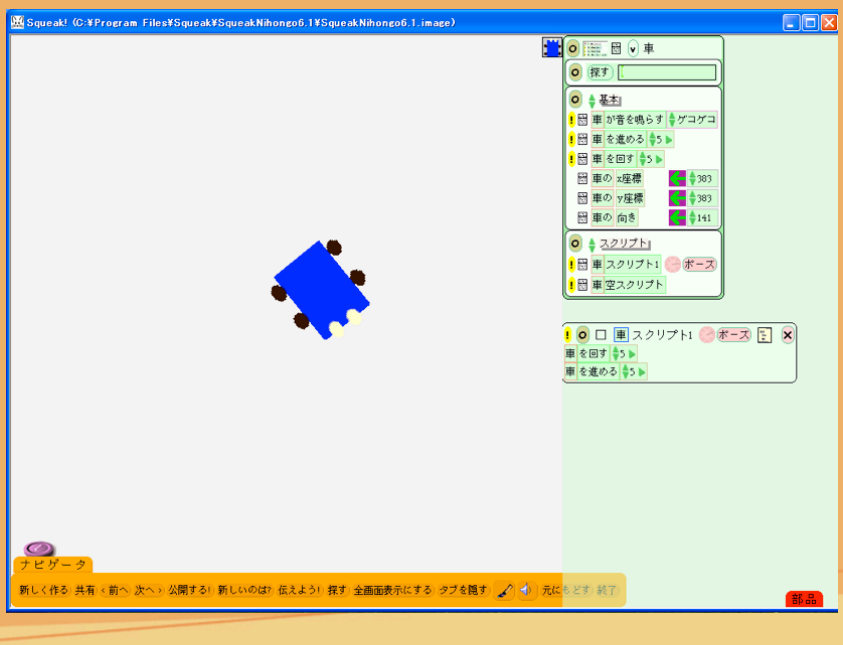
コントロールツール “ハロ” の説明



プロジェクト

- ~~車を作ろう~~
- 車を運転しよう
- 利口な車を作ろう

車を運転しよう

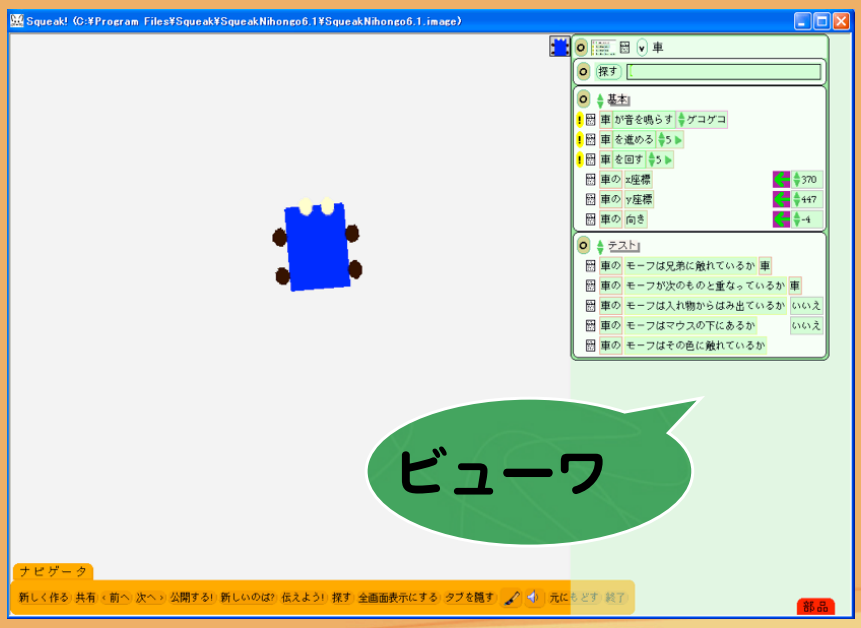


自分のビューワを開く

（2）
ここをクリック

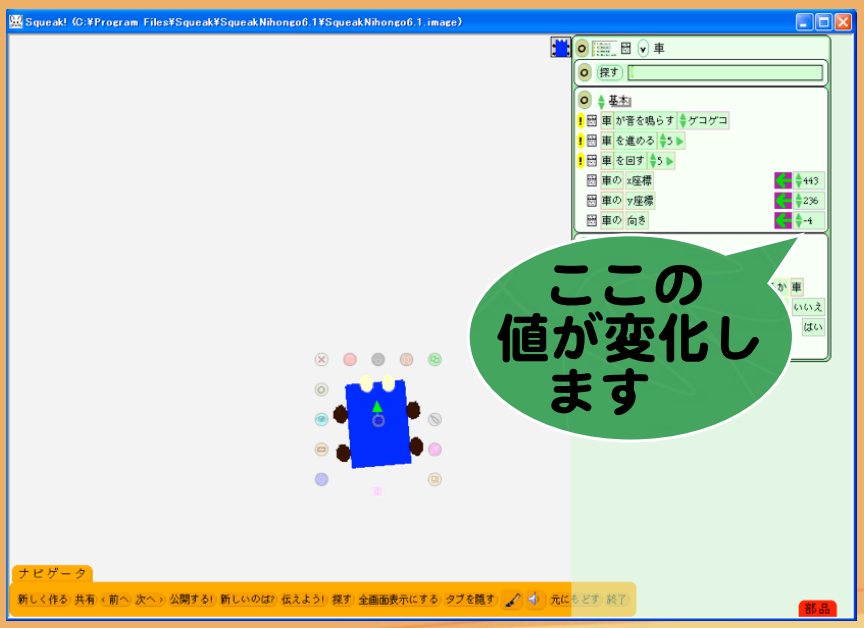
（1）Alt+左クリック

自分のビューワを開く

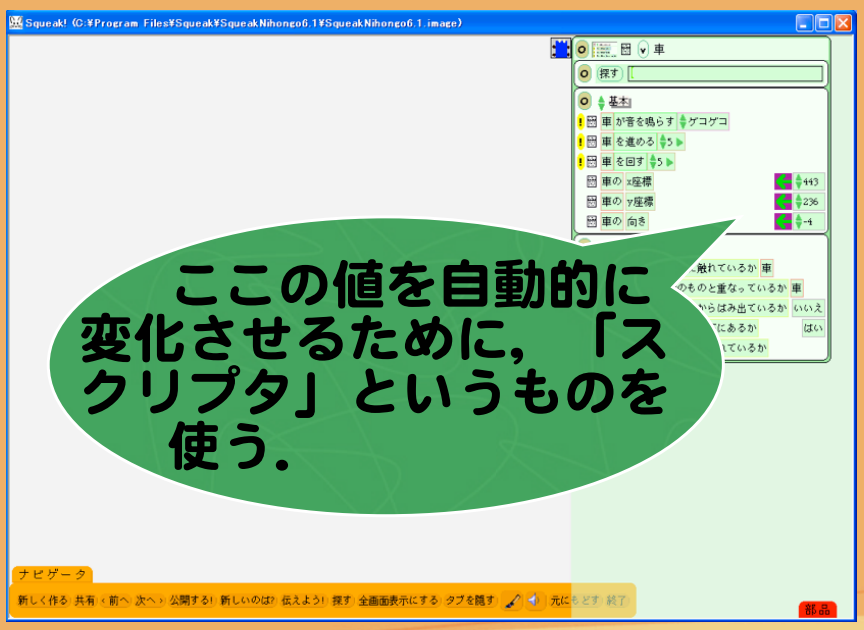


ビューワ

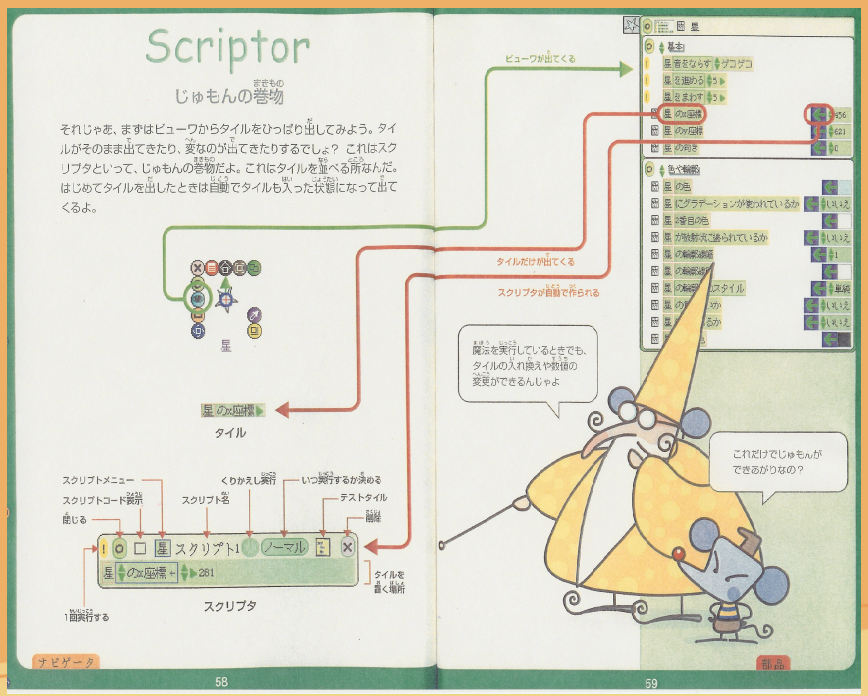
車を引っ張りまわすと…



車を自力で動かすには

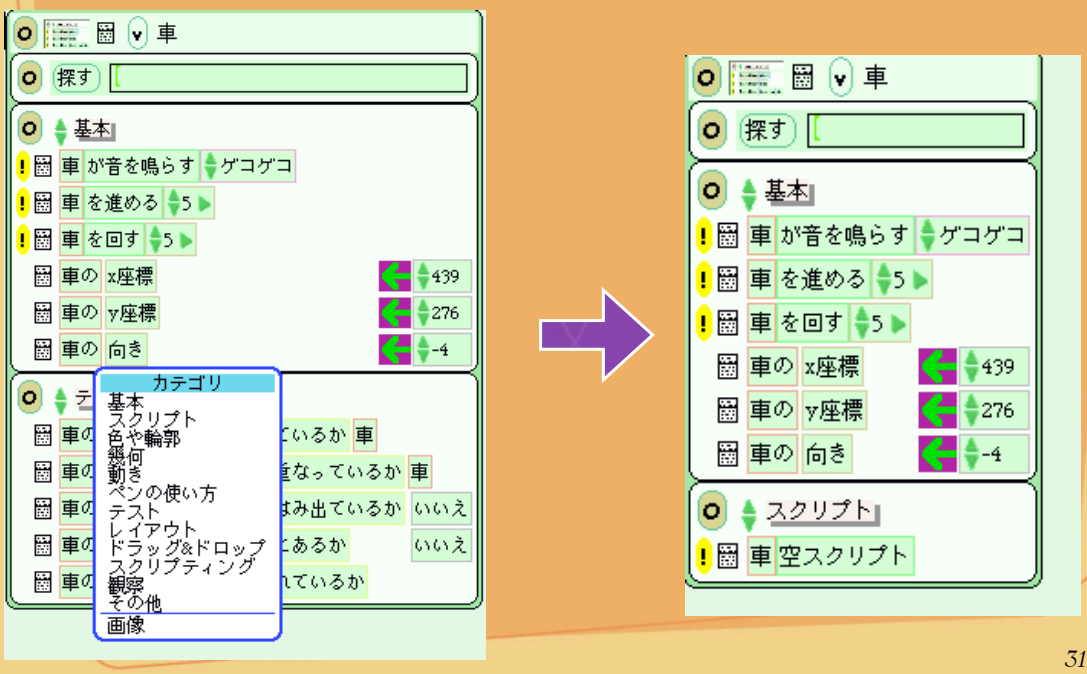


スクリプタ

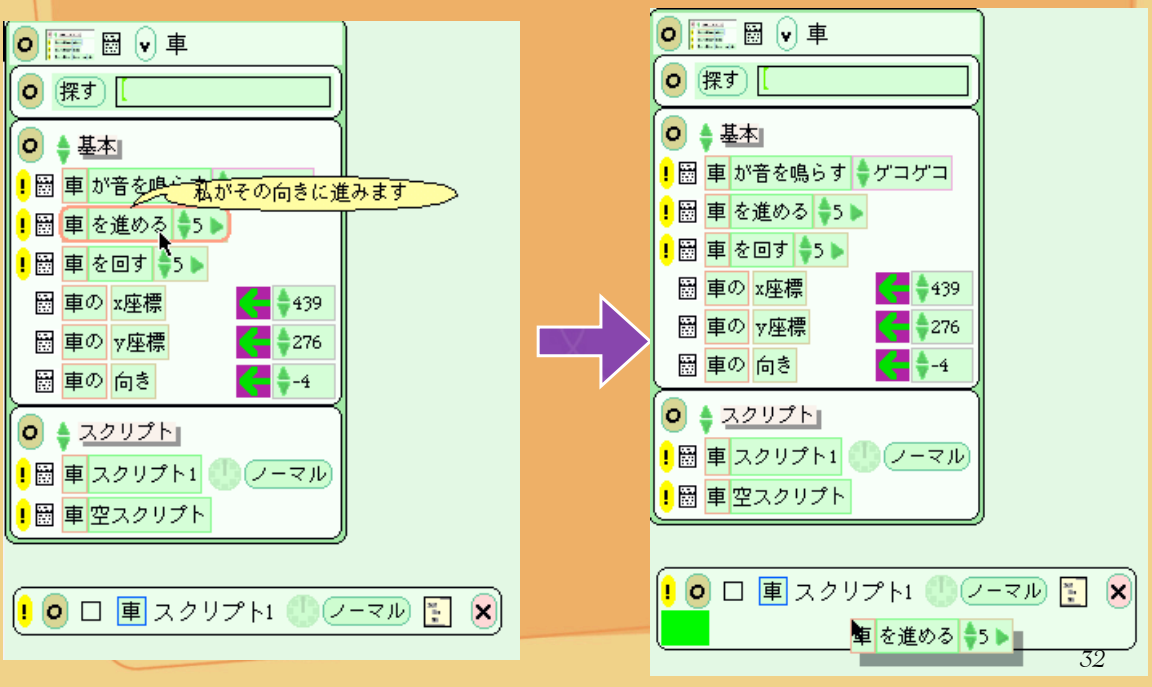


運転する手順を決める

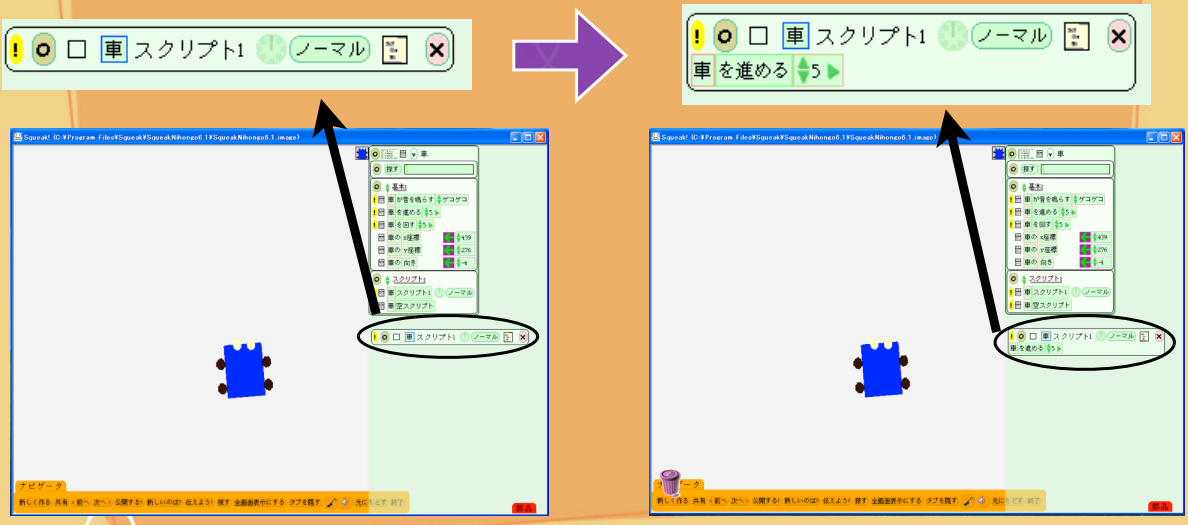
スクリプタの表示



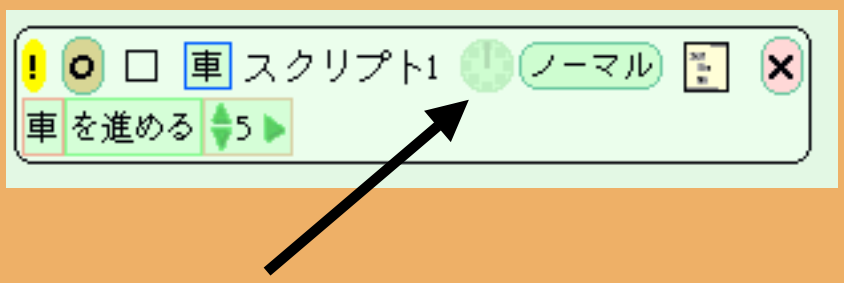
スクリプトの編集



スクリプトの編集の前後

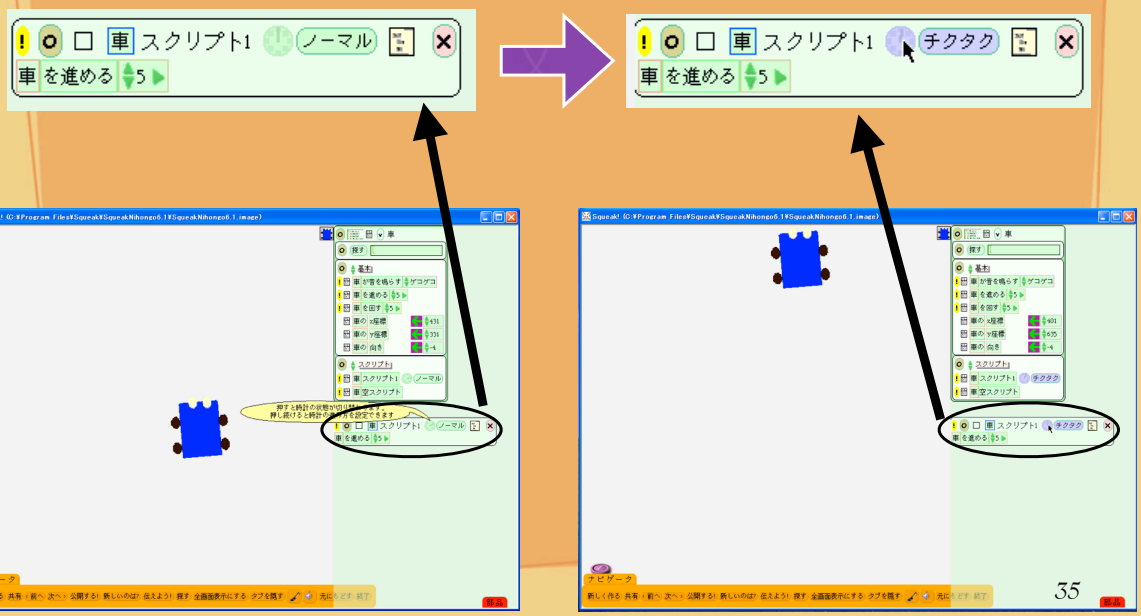


スクリプトを動かす

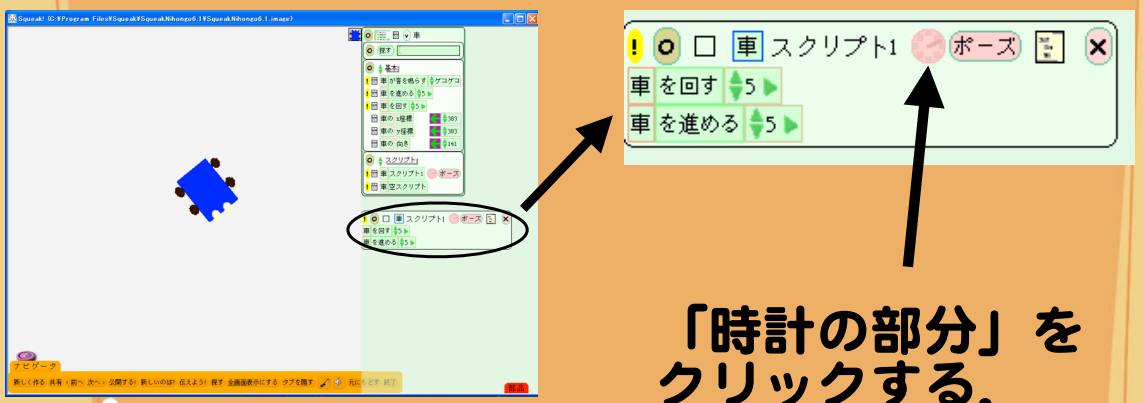


「時計の部分」をクリックすると，
スクリプトが動きます．

車を前進させる



車を回転させ前進させる

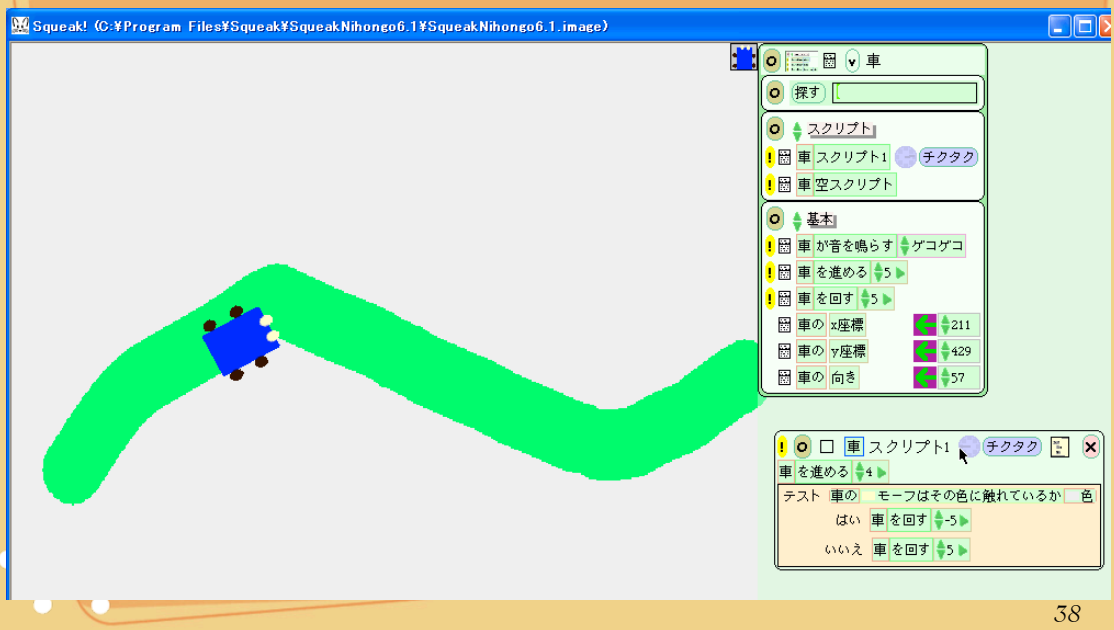


「時計の部分」を
クリックする.

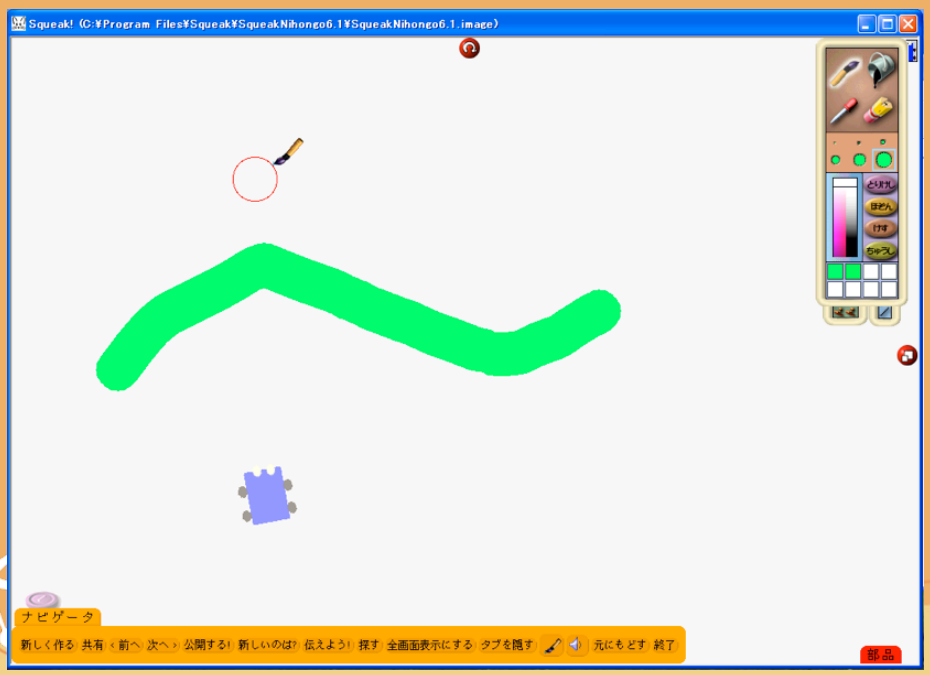
プロジェクト

- ~~車を作ろう~~
- ~~車を運転しよう~~
- 利口な車を作ろう

利口な車を作ろう



準備：コースを作る



コースに沿って車を走らせてみよう

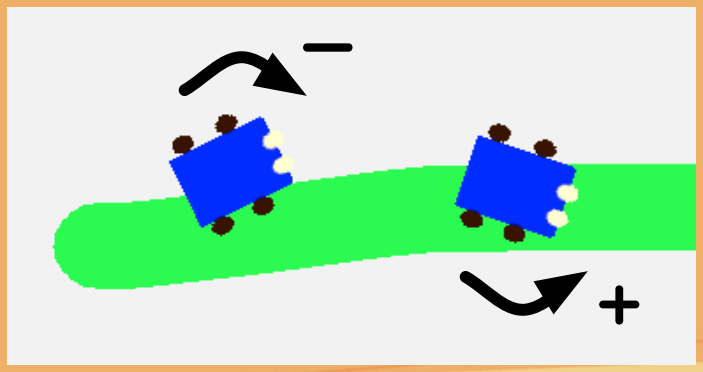
- スクリプト１の「車を回す」の値を変化させて，車をコースに沿って走らせてみよう．
  - コツ：「車を進める」の値を小さくする．

自動的にコースに沿って車を走らせてみよう

- コースから外れないように「自動的」に車の向きを調整する.
- 「車の向きを調整する」ためのルールを決める.

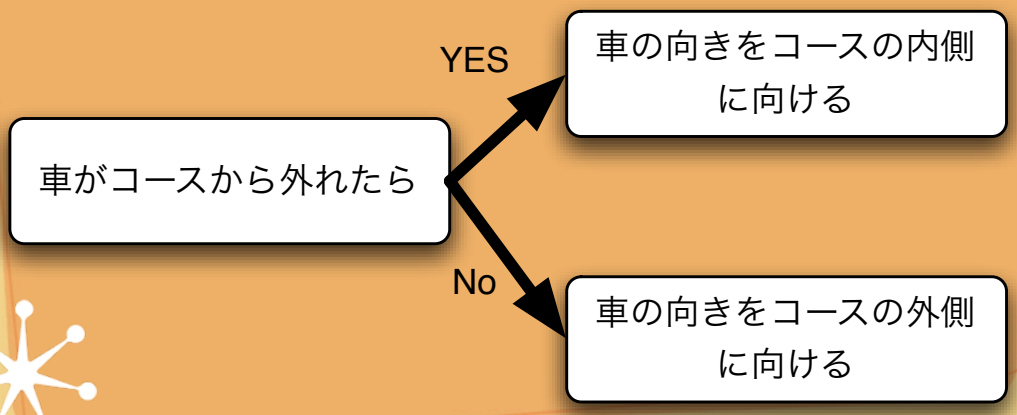
車の向きの調整ルール：1

- 車がコースから外れたとき，車の向きをコースの内側に向けるようにする.



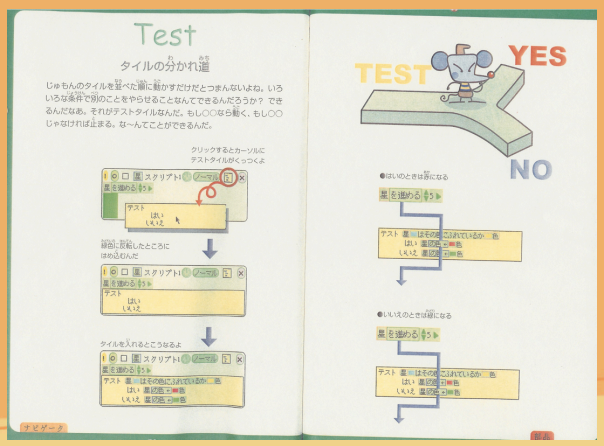
車の向きの調整ルール：2

- 調整ルールは，「別れみち」になっている．

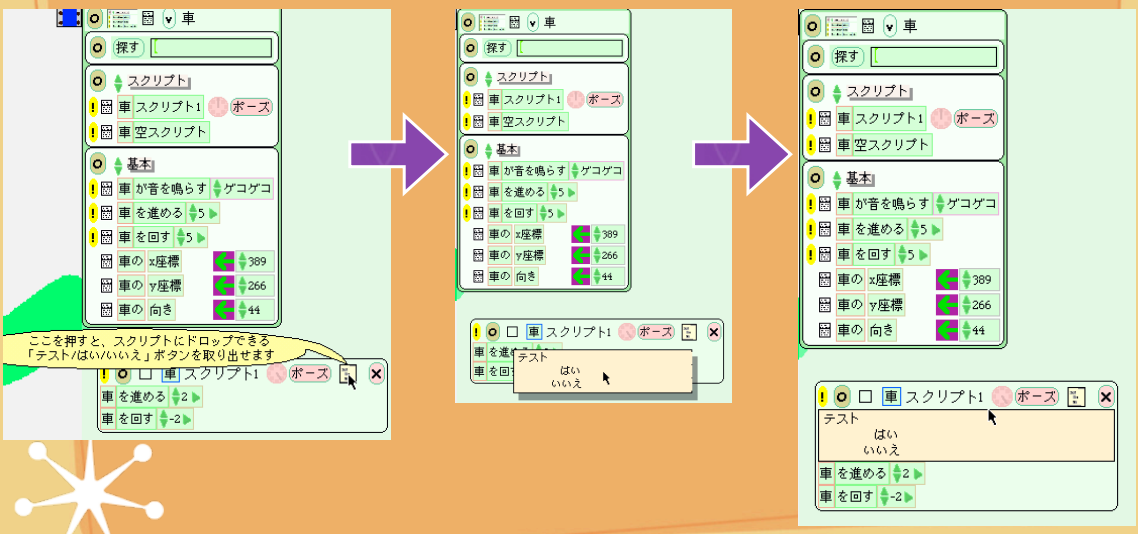


「Test（テスト）」

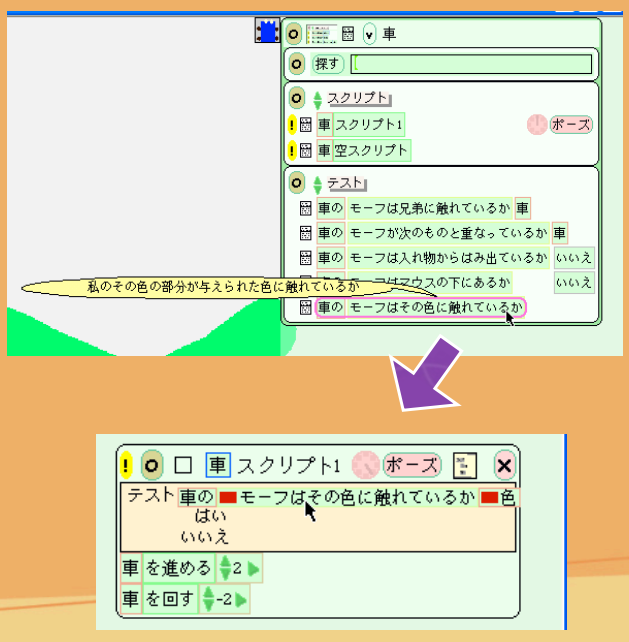
- 調整ルールを作るために，「テスト」というものを使う．



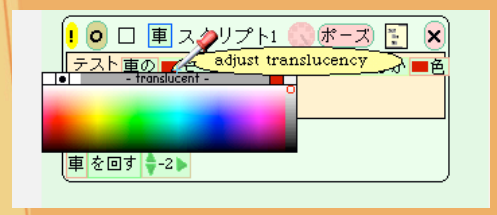
「テスト」の代入：1

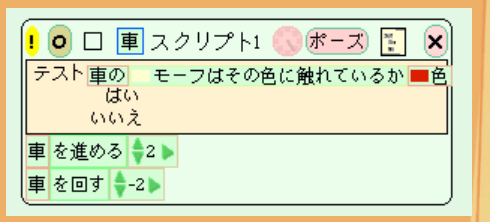


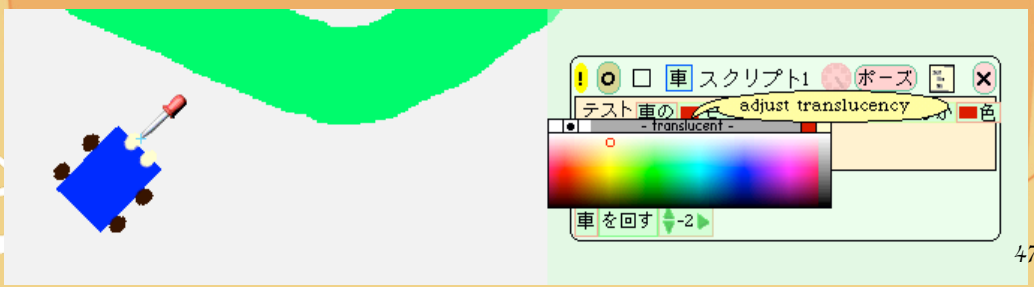
「テスト」の代入：2



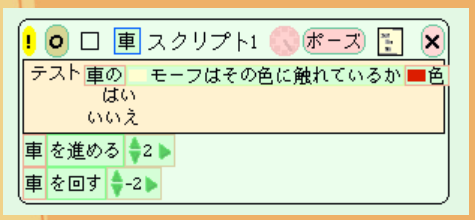
調整ルールの作成：1

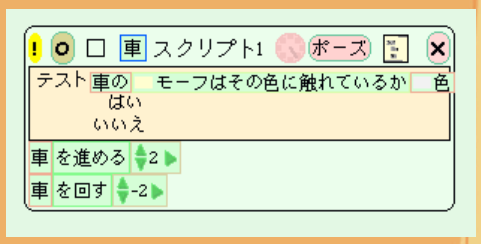




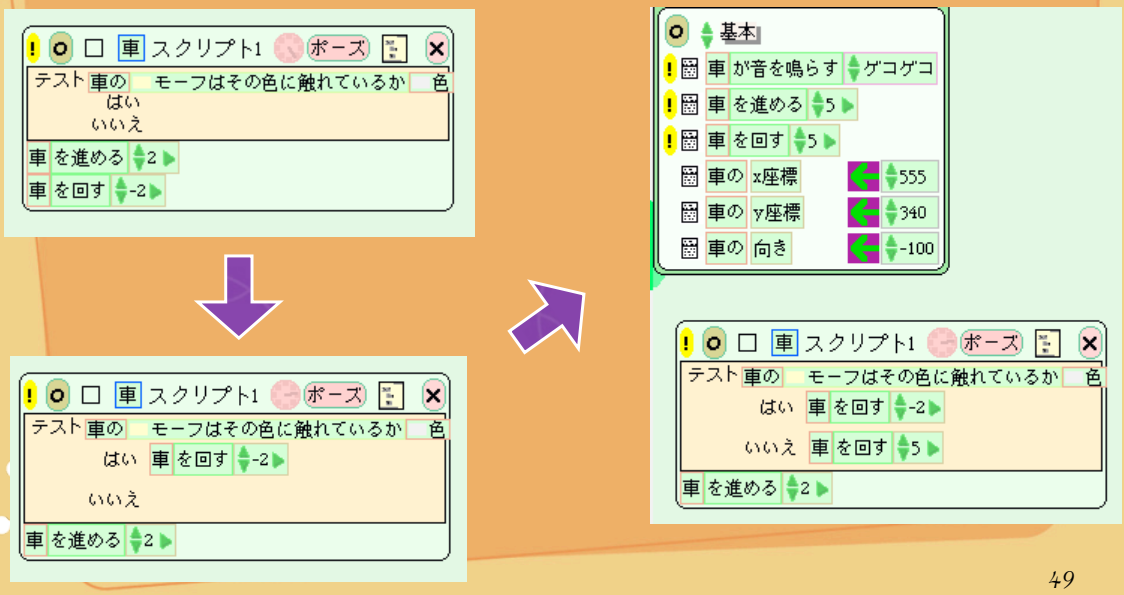


調整ルールの作成：2

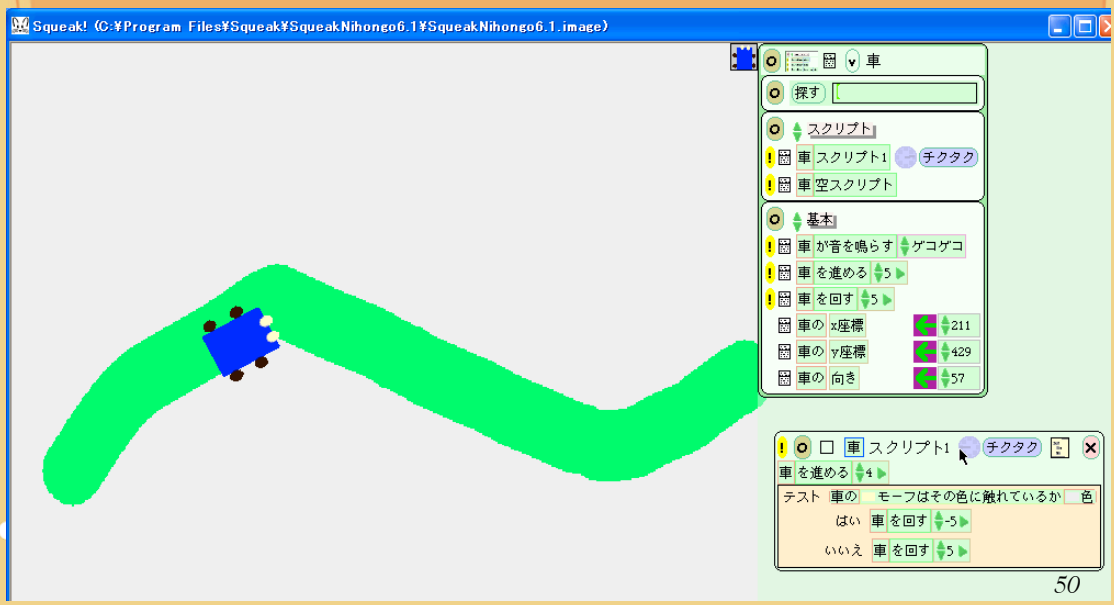




調整ルールの作成：3



調整ルールに基づいて，自動車を動かす

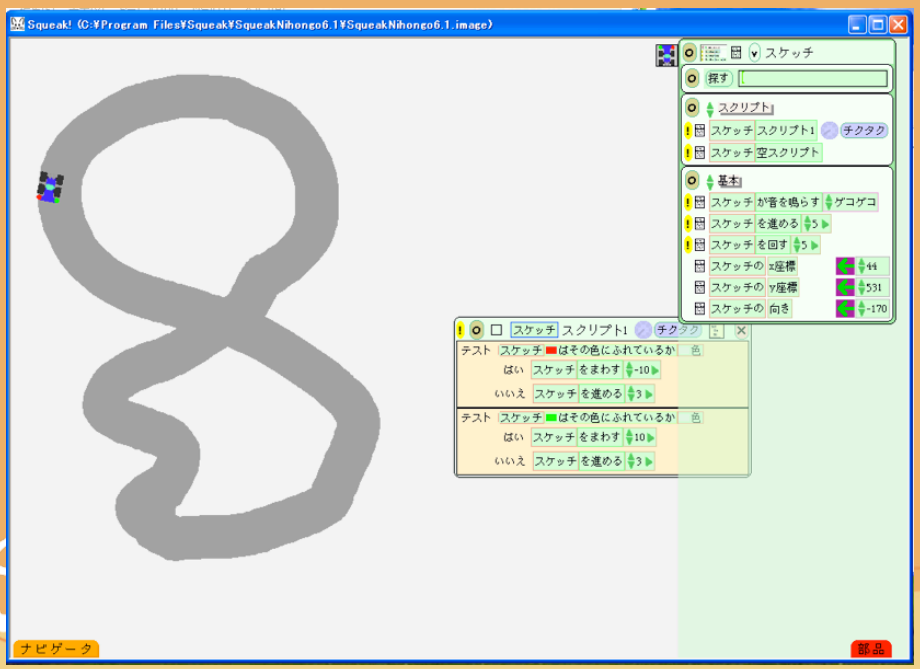


自動制御：フィードバック制御

- 調整ルールに基づいて，何かを制御する．
- ここでは，車の方向を制御している．
- 例：クーラーの温度調整も同じ原理

詳しい内容は，大学の２年生ぐらいで勉強する．

もっと利口な車



まとめ

- 「スクイーク」を使って，「コンピュータを操る方法（プログラミング）」を実習しました．
  - コンピュータの画面で，車を制御するソフトを作成した．

質問

？

hideaki@jaist.ac.jp