

Red/View□□□□□□□□□□

□□

1. □□□□□□	2
2. □□□□□□□□	2
2.1. □□□□□□□□	4
3. □□□□□□□□	4
4. Para □□□□□□	5
5. □□□□□□	5
6. □□□□□□□□	5
6.1. Base	5
6.2. Text	6
6.3. Button	6
6.4. Check	7
6.5. Radio	7
6.6. Field	7
6.7. Area	8
6.8. Text-list	9
6.9. Drop-list	9
6.10. Drop-down	10
6.11. Progress	10
6.12. Slider	10
6.13. Camera	11
6.14. Panel	11
6.15. Tab-panel	11
6.16. Window	12
6.17. Screen	13
6.18. Group-box	13
7. □□□□□□□□□□	13
8. SHOW□□	13
9. □□□□□□ VS □□□□□□□□	14
10. □□□□□□□□□□	15
11. □□□□	15
11.1. □□□□□□	15
11.2. Event! □□□□	17
11.3. Actors	18
11.4. □□□□□□□□	19
11.5. □□□□□□□□□□□□	20
11.5.1. insert-event-func	20

11.5.2. remove-event-func	20
12. System/view object	21
13. View.....	21
14.	21

1.

Red/View.....View.....Red.....

-API
-
-
-
-
- OS.....
-

.....

2.

..... **face!** *(facet)*

.....:

.....
type	word! (.....)
offset	pair!
size	pair!
text	string!
image	image!
color	tuple!	R.G.B.....R.G.B.A.....
menu	block!
data	any-type!
enable?	logic!
visible?	logic!
selected	integer!

変数名	型	初期値	デフォルト値	説明
flags	block!, word!	0	0	フラグのビット列。初期値は0。
options	block!	0	0	[name: value]の辞書。初期値は空辞書。
parent	object!	0	0	親オブジェクトの参照。初期値は0。
pane	block!	0	0	パネの参照。初期値は0。
state	block	0	0	状態の参照。初期値は0。
rate	integer!, time!	0	0	整数型integerと時間型timeの参照。初期値は0。
edge	object!	0	0	エッジの参照。初期値は0。
para	object!	0	0	パラメータの参照。初期値は0。
font	object!	0	0	フォントの参照。初期値は0。
actors	object!	0	0	アクターの参照。初期値は0。
extra	any-type!	0	0	任意の型の参照。初期値は0。
draw	block!	0	0	描画の参照。初期値は0。

flags の初期値は0:

変数名	型
all-over	0 over 初期値は0

初期値は0の初期値は0

Notes:

- 初期値は0の初期値は0 none 初期値は0
- offset と size 初期値は0
- offset と size 初期値は0 none 初期値は0(初期値は0) 初期値は0
- 初期値は0 color、image、text、draw 初期値は0

初期値は0face!初期値は0初期値は0 初期値は0 type 初期値は0

button: make face! [type: 'button']

初期値は0 type 初期値は0

2.1. □□□□□□□□

□ □

drag-on	'down' mid-down alt-down aux-down &

3. □□□□□□□□

□ □ □ □ □ □ □ □ □ □

font

[illegible]

1010101010

項目名	型別	単位	説明
name	string!	文字列	OS固有のファイル名
size	integer!	文字数	ファイルのサイズ
style	word!, block!	文字列	ファイルのスタイル
angle	integer!	角度	ファイルの角度
color	tuple!	色	R.G.BまたはR.G.B.Aの値
anti-alias?	logic!, word!	文字列	アンチエイリアス
shadow	(reserved)	文字列	シャドウ
state	block!	文字列	ファイルの状態
parent	block!	文字列	ファイルの親

Notes:

- `None` is a special value representing the absence of a value
- `angle` is a special value representing the angle of a vector
- `...` is a special value representing the end of a sequence

[illegible]

- **bold**
- *italic*
- underline
- ~~strike~~

[illegible]

- 抗锯齿 (anti-alias?: yes/no)
- 抗锯齿类型 (anti-alias?: 'ClearType')

4. Para 属性

Para属性

para!

Para属性用于设置HTML元素的默认样式。1. 设置元素的默认样式。2. 设置元素的默认样式。

属性	值	说明
origin	属性	设置元素的默认样式
padding	属性	设置元素的默认样式
scroll	属性	设置元素的默认样式
align	word!	设置元素的默认样式: left, center, right
v-align	属性	设置元素的默认样式: top, middle, bottom
wrap?	logic!	设置元素的默认样式
parent	block!	设置元素的默认样式

Notes:

- Para属性: none 设置元素的默认样式

5. Pane 属性

Pane属性用于设置HTML元素的默认样式。

- pane 属性: 设置元素的默认样式
- parent 属性: 设置元素的默认样式

pane 属性: 设置元素的默认样式。Z 属性: 设置元素的默认样式。

pane

screen 属性: 设置元素的默认样式。pane 属性: 设置元素的默认样式。

screen 属性: 设置元素的默认样式。window 属性: 设置元素的默认样式。

[Face tree] | [face-tree.png](#)

6. Base 属性

6.1. Base

base 属性: 设置元素的默认样式。base 属性: 设置元素的默认样式。

Facet	值
type	'base'

options	default
flags	block!

options

- **no-border** GUI

data text

- **text** data load text none options/default
- **data** text data form

options

- **default** text none data text

NOTE:

- **selected**

enter	on-enter	Enter
change	on-change	
key	on-key	

6.7. Area

options

Facet	
type	'area'
text	
flags	area block!

options

- **no-border** GUI

Notes:

- **selected**
- A vertical scroll-bar can appear if all lines of text cannot be visible in the area (might be controlled by a **flag** option in the future).
- **area** flag

Facet	Value	Notes
change	on-change	When the value of the facet changes, the selected value is updated.
key	on-key	When the key of the facet changes, the selected value is updated.

6.8. Text-list

The text-list facet is used to display a list of text values. It is a simple list of text values, and it is used to display a list of text values.

Facet	Value	Notes
type	'text-list'	
data	block! hash!	
selected	none	

Facet	Value	Notes
select	on-select	When the value of the facet changes, the selected value is updated.
change	on-change	When the value of the facet changes, the selected value is updated.

Notes:

- The text-list facet is used to display a list of text values.

6.9. Drop-list

The drop-list facet is used to display a list of text values. It is a simple list of text values, and it is used to display a list of text values.

Facet	Value	Notes
type	'drop-list'	
data	block! hash!	
selected	none	

data block! hash!
selected 1 block! hash!
data block! hash!

Facet	Value	Notes
select	on-select	When the value of the facet changes, the selected value is updated.
change	on-change	When the value of the facet changes, the selected value is updated.

Notes:

- `selected` 属性用于指定当前选中的选项

6.10. Drop-down

下拉菜单（Drop-down）是一种常见的UI元素，用于展示一组选项，并允许用户从中选择一个值。

Facet	属性
<code>type</code>	<code>'drop-down'</code>
<code>data</code>	选项列表，格式为 <code>block! hash!</code>
<code>selected</code>	当前选中的选项的索引，默认为 <code>none</code>

`data` 属性用于指定下拉菜单的选项列表。选项列表的格式为 `block! hash!`，其中 `block!` 表示选项的文本，`hash!` 表示选项的值。例如，`data: ['option 1', 'option 2', 'option 3']`。

属性	事件	说明
<code>select</code>	<code>on-select</code>	当用户选择一个选项时触发。返回选中的选项的索引。
<code>change</code>	<code>on-change</code>	当用户更改选项时触发。返回选中的选项的索引。

Notes:

- `selected` 属性用于指定当前选中的选项

6.11. Progress

进度条（Progress）是一种常见的UI元素，用于显示任务的完成进度。

Facet	属性
<code>type</code>	<code>'progress'</code>
<code>data</code>	进度值，格式为 <code>percent!float!</code>

Notes:

- `float` 属性 `data` 用于指定进度值，范围从 `0.0` 到 `1.0`

6.12. Slider

滑块（Slider）是一种常见的UI元素，用于允许用户通过拖动来调整数值。

Facet	属性
<code>type</code>	<code>'slider'</code>
<code>data</code>	滑块值，格式为 <code>percent!float!</code>

Notes:

- float **data** 0.0 1.0

6.13. Camera

Facet	
type	'camera'
data	
selected	data none

Notes:

- data none show
- to-image

6.14. Panel

Facet	
type	'panel'
pane	Z

Notes:

- offset
-

6.15. Tab-panel

Facet	
type	'tab-panel'
data	
pane	block!
selected	noneinteger!

6.17. Screen

Facet	□□
type	'screen'
size	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□pair!□
pane	□□□□□□□□□□□□□□□□□□block!□

6.18. Group-box

[illegible]

Facet	
type	'group-box
pane	000000000000000000000000Z00000000000

Notes:

- `offset` 偏移量
- 偏移量

7. □□□□□□□□□□

- [illegible]

Notes:

- free

8. SHOW□□

show <face>

11

state
pane

```

state
OS API

```

Notes:

- show **changes** **deferred**
- OS **handle!**

9. 〇〇〇〇〇〇 VS 〇〇〇〇〇〇〇〇〇

[illegible][illegible]

- `show`
-
-

- [illegible]

- `Rebol/View`

```

series Red

```

API series

```
view [
  list: text-list data ["John" "Bob" "Alice"]
  button "Add" [append list/data "Sue"]
  button "Change" [lowercase pick list/data list/selected]
]
```

11.1. □□□□□

15

名前	タイプ	説明
drop	イベント	ドロップイベント
click	イベント	クリックイベント
dbl-click	イベント	ダブルクリックイベント
over	イベント	マウスが要素の上に移動したときに発生するイベント。2つの イベント flags イベント all-over イベント
move	イベント	移動イベント
resize	イベント	リサイズイベント
moving	イベント	移動中イベント
resizing	イベント	リサイズ中イベント
wheel	イベント	ホイールイベント
zoom	イベント	ズームイベント
pan	イベント	パンイベント
rotate	イベント	回転イベント
two-tap	イベント	2回タップイベント
press-tap	イベント	押しタップイベント
key-down	イベント	キーダウンイベント
key	イベント	キーイベント。control、shift、menu
key-up	イベント	キーアップイベント
enter	イベント	Enter
focus	状態	フォーカス状態
unfocus	状態	アンフォーカス状態
select	状態	選択状態
change	状態	変更状態
menu	状態	メニュー状態
close	状態	閉状態
time	イベント	時間イベント rate イベント

Notes:

- Windows XP
- 1 **moving** イベント **move** イベント
- 1 **resizing** イベント **resize** イベント

11.2. Event! いろいろ

いろいろなイベントが用意されています。ここでは、いくつかのイベントについて説明します。

イベント名	説明
<code>type</code>	イベントの種類 (word!)
<code>face</code>	イベントの顔 (object!)
<code>window</code>	イベントのウィンドウ (object!)
<code>offset</code>	イベントのオフセット (pair!) 2つの整数のペア
<code>key</code>	イベントのキー (char! word!)
<code>picked</code>	イベントのピッキング (integer! percent! wheel! menu! ID! word! Windows! GESTUREINFO! fullArguments!)
<code>flags</code>	イベントのフラグ (block!)
<code>away?</code>	イベントが遠ざかっているかどうか (true logic! over)
<code>down?</code>	イベントが押されているかどうか (true logic!)
<code>mid-down?</code>	イベントが中クリックされているかどうか (true logic!)
<code>alt-down?</code>	イベントがAltキーが押されているかどうか (true logic!)
<code>ctrl?</code>	イベントがCtrlキーが押されているかどうか (true logic!)
<code>shift?</code>	イベントがShiftキーが押されているかどうか (true logic!)

`event/flags` イベントのフラグ

- `away`
- `down`
- `mid-down`
- `alt-down`
- `aux-down`
- `control`
- `shift`

Notes:

- `type` イベントの種類 (type) イベントの種類 (type)

`event/key` イベントのキー (word!)

- `page-up`
- `page-down`

- end
- home
- left
- up
- right
- down
- insert
- delete
- F1
- F2
- F3
- F4
- F5
- F6
- F7
- F8
- F9
- F10
- F11
- F12

key-down □ key-up □□□□□□□□□□□□□□□□ event/key □□□□□□□□

- left-control
- right-control
- left-shift
- right-shift
- left-menu
- right-menu

11.3. Actors

actors

[illegible]

11

```
on-<event>: func [face [object!] event [event!]]
```

```
<event> : []
```

```
face      : ████████████████████████████████████
```

```
event : [ ]
```

GUI on-create on-create
create face

103

```
'stop' : []
```

```
'done' : []
```

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

11.4. □□□□□□

[illegible]

- [illegible]

[Event flow] | *event-flow.png*

□ □

1. `onmouseover` 事件在鼠标指针移到元素上时触发
2. `onmouseout` 事件在鼠标指针移出元素时触发
 - a. `onmouseover` 事件在鼠标指针移到元素上时触发 `on-detect` 事件
 - b. `onmouseout` 事件在鼠标指针移出元素时触发 `on-detect` 事件
 - c. `onmouseover` 事件在鼠标指针移到元素上时触发 `on-detect` 事件
3. `onmousedown` 事件在鼠标按下时触发
 - a. `onmousedown` 事件在鼠标按下时触发 `on-click` 事件
 - b. `onmousedown` 事件在鼠标按下时触发 ``on-click`` 事件
 - c. `onmousedown` 事件在鼠标按下时触发 `on-click` 事件

Notes:

- `done word`
- `system/view/capturing?: yes`

11.5. 事件管理API

このセクションでは、イベント管理APIについて説明します。

11.5.1. insert-event-func

概要

```
insert-event-func <handler>

<handler> : 関数
            func [face [object!] event [event!]]
```

説明

```
insert-event-func function!
```

例

この関数は、イベント管理システムに新しいイベントハンドラを追加します。ハンドラは、face (オブジェクト) と event (イベント) を引数として呼び出されます。function は、ハンドラの関数型であることを示します。

戻り値

- none : 成功した場合、none! を返します。
- 'done : 成功した場合、word! を返します。
- 'stop : 失敗した場合、word! を返します。

関連関数

11.5.2. remove-event-func

概要

```
remove-event-func <handler>

<handler> : 関数
```

説明

この関数は、イベント管理システムから指定されたイベントハンドラを削除します。

12. System/view object

Word	説明
screens	画面のリスト
event-port	イベントポート
metrics	メトリクス
platform	プラットフォーム
VID	VIDオブジェクト
handlers	ハンドラー
reactors	リアクター
evt-names	イベント名
init	初期化
awake	目覚め
capturing?	yes = 検出 detect している no していない
auto-sync?	yes = 自動同期 no = 手動同期
debug?	yes = デバッグ on no していない
silent?	yes = VIDDraw しない no する

13. Viewオブジェクト

Viewオブジェクトは、プラットフォーム、イベントポート、メトリクス、Redオブジェクト、Needsオブジェクトを持つ。

```
Red [
  Needs: 'View'
]
```

NOTE: red オブジェクトは、Viewオブジェクトを持つ。Needs

14. 関数

Function	説明
view	VIDオブジェクトを /no-wait で表示する
unview	VIDオブジェクトを非表示にする
layout	VIDオブジェクトのレイアウトを設定する

Function	関数
center-face	顔の中心に配置する
dump-face	顔のデータをダンプする
do-actor	アクターの動作を実行する
do-events	イベントを実行する
draw	Draw関数でimageを描画する
to-image	faceをimageに変換する
size-text	テキストのサイズを設定する

関数の一覧

- 関数の一覧
- image!関数