

#MASTER BARで質問している のルーチン。NPC と遭遇することができずエラーに (泣

Sub Change Condition

sub change_condition {

my \$S = shift;

my \$T = shift; # da,ij,cd

my \$V = shift; # 状態略字

my \$F = shift; # true -> 追加 , false -> 削除

```
    if ($F) {
        if ($T eq 'cd') {
            $S->{Cd}{$V} = $set::cnd{$V}[1] if $V;
        }
        else {
            $S->{$T} = $V;
        }
    }
    else {
        if ($T eq 'cd') {
            undef $S->{Cd}{$V} if $V;
        }
        elsif ($T eq 'ij') {
            undef $S->{ij};
        }
        else {
            $S->{da} = 'Av';
            $S->{Da} = $set::cnd{$S->{da}}[1];
            $S->{lf} = 1;
            &reload_gage('lf','V');
        }
    }
}
```

return if \$S->{id} ne \$F{id};

&condition_initial(\$S,\$T,1);

}

Sub Condition_initial

sub condition_initial {

my \$S = shift;

my \$T = shift; # da,ij,cd

my \$F = shift; # true -> reload

my (@L,\$K,\$V);

```
    if ($T eq 'da') {
        $S->{Da}{in} = $set::cnd{$S->{da}}[0];
        $V = $set::cnd{$S->{da}}[1];
    }
    elsif ($T eq 'ij') {
        $V = $S->{ij}{in} = $set::cnd{$S->{ij}}[0];
        undef $S->{ij} if !$V;
    }
    else {
        delete $S->{Cd}{in};
        while (($K,$V) = each %{$S->{Cd}}) {
            push(@L,$set::cnd{$K}[0]) if $V;
        }
        @L = sort @L;
        $V = $S->{Cd}{in} = join('・',@L);
    }
}
```

```

        undef $S->{cd} if !$V;
    }

    return if !$F;

    &reload_data($T,$V);

}

# Sub Get Equipment #
sub get_equipment {
    my $S = shift;
    my $E = shift;
    my @L;

    @L = split(/¥/, $S->{$E});
    $E = ucfirst $E;
    $S->{$E}{nm} = $L[0]; # Name
    $S->{$E}{cl} = $L[1]; # Class
    map { $S->{$E}{ql}{$_} = 1 } split(/,/,$L[2]); # Quality
    $S->{$E}{ef} = $L[3]; # Effect
    $S->{$E}{db} = $L[4]; # Durability
    $S->{$E}{im} = $L[5]; # Image
    $S->{$E}{gd} = $L[6]; # Grade
    $S->{$E}{pl} = $L[7]; # Plus
    $S->{$E}{cr} = $L[8]; # Creator
    $S->{$E}{lm} = &item_image($S->{$E}); # Image with Tag

}

# Sub Change Status #
sub change_status {
    my $S = shift;
    my $T = shift; # ステータス略字
    my $V = shift; # 増減させる値
    my $F = shift; # true -> メッセージを表示する
    my $X = shift; # Max 値が他にある場合
    my ($Y,$Z);

    $V += 1 if $T eq 'vt' && $S->{Cd}{En} && $V < 0;

    $S->{$T} += $V;
    $Y = $T eq 'hl' ? 'Mn':
        $T eq 'ap' ? 'Dt':
        $T eq 'hy' ? 'En':
        ' ';

    if ($S->{$T} <= 0) {
        $S->{$T} = 0;
        if ($Y && !$S->{Cd}{$Y}) {
            &change_condition($S,'cd',$Y,1);
            $Z = "$set::cnd{$Y}[1] になりました。 ";
        }
    }
    elsif ($S->{"M$t"} && $S->{$T} >= $S->{"M$t"} && !$X) {
        $S->{$T} = $S->{"M$t"};
        if ($Y && $S->{Cd}{$Y}) {

```

```

        &change_condition($S,'cd',$Y);
        $Z = "$set::cnd{$Y}[1] が治りました。";
    }
}
elseif ($X && $S->{$T} > $X) {
    $S->{$T} = $X;
}

return if !$F;

if ($V > 0) {
    &M("<span class=up>$set::lab{$T} が<span class=alert>$V</span> 上昇 </span>");
}
else {
    $V = abs($V);
    &M("<span class=down>$set::lab{$T} が<span class=alert>$V</span> 減少 </span>");
}

&M("<img src=$set::mim_dir/redcross.gif> <span class=sick>$Z</span>") if $Z;

}

# Sub Item Image #
sub item_image {
my $V = shift;

    return "<img src=$set::iim_dir/$V->{im}> " if $V->{im};

}

# Sub Sort Item #
sub sort_item {
my $S = shift;

    require "$set::dat_dir/sort.cgi";

    @{$S->{Bg}} = sort { $S0{$a->{cl}} <=> $S0{$b->{cl}} || $a->{id} cmp $b->{id} || $b->{qn} <=>
    $a->{qn} } @{$S->{Bg}};

}

# Sub Roll Dice #
sub roll_dice {
my $dice = shift;
my $plus;
my $roll;
my $V;

# srand(time|$S) if !$diceflag;

    return $' if $dice = s/^S//;
    ($dice,$plus) = split(/¥+/, $dice);
    ($roll,$dice) = split(/D/, $dice);

```

```

        if (!$dice) {
            $dice = $roll;
            $roll = 1;
        }
        if ($dice < 2) { return 1 }
        while ($roll--) {
            $V += int(rand($dice)) + 1;
        }
        $V += $plus;
        $V = 1 if $V < 1;

# $diceflag = 1;

        return $V;

    }

# Sub M #
sub M {
    my $M = shift;
    my $break = shift; # value = 1;

    $break = $break == 1 ? '' : "<br>¥n";
    push(@{$!->{M}}, "$M$break");
    return 1;

}

# Sub Select Acquisition #
sub select_acquisition {
    my $D = shift;
    my $i = 1;
    my (@L,$V);

    while ($D->{"i$i"}) {
        push(@L,$D->{"i$i"});
        $i++;
    }

    $V = &roll_dice($#L + 1) - 1;

    return $L[$V];

}

# Sub Get Item #
sub get_item {
    my $S = shift;
    my $T = shift;
    my ($V,$i,$j,@L,@K);

    $V = $T == 2 ? $S->{op} : $S->{Bg};

```

```

    @L = map { chomp $_; [split(/<>/)] } @V;

    $V = ¥@set::idt;

    foreach $i (0 .. $#L) {
        foreach $j (0 .. $#V) {
            $K[$i][$V->[$j]] = $L[$i][$j];
        }
        map { $K[$i][Ql][$_] = 1 } split(/,/,$K[$i][ql]);
    }

    $S->{Bs} = @K if $T == 0;
    $S->{Bk} = @K if $T == 3;

    $S->{Bg} = [@K];

}

# Sub Set Item #
sub set_item { }

# Sub Reload Data #
sub reload_data { }

# Sub Reload Gage #
sub reload_gage { }

# Sub SOS2 to SOS3 #
sub SOS2toSOS3 {
    my $V = shift || "";
    my $S;
    my $T;
    my ($i,$MG);
    my ($Si,$item,$sts,$price,$ipc,$iak,$idd);

    # 魔法選択 #
    if (${$V.'sp'} = /Mg/) {
        require "$set::dat_dir/magic.cgi";
        for (1 .. $set::ffx) {
            $i = int(rand ${$V.'ak'}+${$V.'ab'});
            $i = $MG if $i > $MG;
            push(@{$MG}, $i);
        }
    }

    $S->{id} = ${$V.'id'};
    $S->{nm} = ${$V.'nm'};
    $S->{ps} = ${$V.'ps'};
    $S->{ip} = ${$V.'ht'};
    $S->{im} = ${$V.'ig'};
    $S->{sx} = ${$V.'sx'} || 0;
    $S->{kg} = 'ven';
    $S->{pl} = 'qui';
    $S->{jb} = ${$V.'jb'};
    $S->{sk} = join(',',split(/(=[A-Z])/,${$V.'sp'})); # スキル

    # ID
    # 名前
    # パスワード
    # IP
    # イメージ
    # 性別
    # 所属国
    # 現在場所
    # 職業
    # Skill

    Identification
    Name
    Password
    Host IP
    Image
    Sex
    Kingdom
    Place
    Job
    Skill

```

```

$$S->{lv} = ${$V.'lv'} || 1; # レベル Level
$$S->{ex} = ${$V.'ex'}; # 経験値 Experience
$$S->{gl} = ${$V.'mn'}; # 所持金 Gold
$$S->{lf} = join('/',${$V.'hp'},${$V.'xp'}); # 生命值 (LIF) Life
$$S->{tc} = ${$V.'dd'}; # 技能 (TEC) Technique
$$S->{st} = ${$V.'ak'}; # 腕力 (STR) Strength
$$S->{dx} = ${$V.'dd'}; # 敏捷 / 器用さ (DEX) Dexterity
$$S->{in} = ${$V.'ak'}; # 知性 (INT) Intelligence
$$S->{cn} = ${$V.'dd'}; # 体力 (CON) Constitution
$$S->{vt} = join('/',${$V.'hp'},${$V.'xp'}); # 気力 (VIT) Vitality
$$S->{hl} = join('/',${$V.'hp'},${$V.'xp'}); # 健康 (HEA) Health
$$S->{ap} = join('/',${$V.'hp'},${$V.'xp'}); # 外見 / 清潔 (APP) Appearance
$$S->{hy} = join('/',${$V.'hp'},${$V.'xp'}); # 幸福 (HAP) Happiness
$$S->{da} = ${$V.'av'} eq 'dead' ? 'Dd' : ${$V.'av'} eq 'stone' ? 'St' : 'Av'; # 生死
Dead or Alive
$$S->{ij} = ''; # 怪我 Injury
$$S->{cd} = ${$V.'av'} eq 'poison' ? 'Ps' : ''; # 状態 Condition
$$S->{wn} = ${$V.'wn'}; # 勝利数 Win
$$S->{ls} = ${$V.'lz'}; # 敗北数 Lose
$$S->{dr} = ${$V.'dr'}; # ドロー数 Draw
$$S->{kl} = ${$V.'kl'}; # 殺害数 Kill
$$S->{dt} = ${$V.'dt'}; # 死亡数 Death
$$S->{rt} = 1500; # 戦闘レート Rate
$$S->{rw} = ${$V.'rw'}; # 報奨金 Reward
$$S->{sc} = 1; # スタンス Stance
$$S->{bg} = ${$V.'xi'}; # 所持 (上限) Bag
$$S->{bs} = ${$V.'bs'}; # 荷物スロット Bag Slot
$$S->{gu} = ''; # ギルド Guild
$$S->{fr} = ''; # 友達 Friends
$$S->{en} = ${$V.'lf'}; # 対戦相手 Enemy
$$S->{mg} = join(',',@{$MG}); # 魔法 Magic
$$S->{bt} = 0; # 登録日 Birth
$$S->{su} = ''; # 状態異常 Status Unusual
$$S->{tl} = 0; # タイムリミット Time Limit
$$S->{dd} = 0; # 死亡日 Death Date
$$S->{li} = !$fg || $fg = /B/ ? 'in' : 'out'; # 接続状態 Login
$$S->{w1} = ${$V.'cm'}; # 自己紹介 (簡) Words 1
$$S->{ob} = ${$V.'is'}; # オープ Orb
$$S->{vx} = ${$V.'xi'}; # 販売上限 Vendor Max
$$S->{fc} = '#000000'; # 文字色 Font Color
$$S->{mb} = time + 10; # 手紙の確認時間 Mail Box
$$S->{px} = join('/',0,${$V.'xi'}); # ペット (上限) Pet Max
$$S->{co} = ''; # 共にするペット Companion
$$S->{pc} = ${$V.'xi'}; # バック技能 Packing Skill
$$S->{lb} = 0; # LevelUp ボーナス Level Up Bonus
$$S->{bk} = ${$V.'xi'}; # 銀行の最大保管数 Bank
$$S->{ab} = ''; # 乗船 Aboard
$$S->{vp} = 1; # 対戦受付フラグ Versus Player
$$S->{dg} = ''; # ダンジョンデータ Dungeon Data

```

```
map { $$S->{Cd}{$_} = $set::cnd{$$_}[1] } split(/,/,$S->{cd});
```

```

$$S->{Da} = $set::cnd{$S->{da}}[1];
$$S->{Ij} = $set::cnd{$S->{ij}}[1];

```

```
map { $$S->{Sk}{$_} = 1 } split(/,/,$S->{sk});
```

```

foreach (@set::sdt,@set::gva) {
    ($S->{$_},$S->{"M$_"}) = split(/¥//,$S->{$_},2);
}

```

```
$T = ¥@set::idt;
```

```

for (split(/,/,${$V.'bg'})) {
    ($item,$sts,$price) = split(/ /);
}

```

```

$ipc = int(substr($sts,0,4));
$iak = int(substr($sts,2,2));
$idd = int(substr($sts,4,2));
$sts = substr($sts,8);

```

```

    $Si->{nm} = $item;
    $Si->{ql} = join(' ', split(/(?=[A-Z])/,$sts));
    $Si->{im} = '_pal.gif';
    $Si->{qn} = 1;
    $Si->{bp} = $ipc;
    $Si->{gd} = $sts = /Li/ ? 2 : 0;
    $Si->{ef} = "lf=$iak,hl=$idd";
    $Si->{db} = $ipc;
    $Si->{wt} = $ipc;
    $Si->{cp} = "st=$iak,tc=$idd";
    $Si->{eq} = $sts = /Qw/ ? 'rh' : $sts = /Qs/ ? 'lh' : $sts = /Qa/ ? 'bd' : $sts = /Qt/ ?
'nk' : ' ';
    $Si->{pl} = $sts = /Li/ ? 2 : 0;
    $Si->{sp} = $price;

    $S->{$Si->{eq}}
"$Si->{nm})/$Si->{cl})/$Si->{ql})/$Si->{ef})/$Si->{db})/$Si->{im})/$Si->{gd})/$Si->{pl})/$Si->{cr}" if
$Si->{eq};

    push(@{$S->{Bg}}, join('<>', @{$Si}{@set::idt}, "¥n"));
}

if (${$V.'bn'}) {
    $item = ${$V.'bn'};
    ($iak) = ${$V.'bp'} = /(¥d+)/;
    ($sts) = ${$V.'bp'} = /^[A-Za-z]+/;

    $Si->{nm} = $item;
    $Si->{ql} = join(' ', split(/(?=[A-Z])/,$sts));
    $Si->{im} = '_pal.gif';
    $Si->{qn} = 1;
    $Si->{gd} = $sts = /Li/ ? 2 : 0;
    $Si->{ef} = "lf=$iak,hl=$idd";
    $Si->{cp} = "st=$iak,tc=$idd";
    $Si->{pl} = $sts = /Li/ ? 2 : 0;
    $Si->{sp} = $price;

    push(@{$S->{op}}, join('<>', @{$Si}{@set::idt}, "¥n"));
}

return $S;

}

# Sub SOS3 to SOS2 #
sub SOS3toSOS2 {
my $S = shift;
my $V = shift || "";
my @L;
my ($Si,$new,$i,$Y,$Z,$item,$sts,$price,$ipc,$iak,$idd);

    ${$V.'lv'} = $S->{lv};
    ${$V.'ex'} = $S->{ex};
    ${$V.'mn'} = $S->{gl};
    ${$V.'hp'} = $S->{lf};
    ${$V.'xp'} = $S->{lhf};
    ${$V.'av'} = $S->{da} eq 'Dd' ? 'dead' : $S->{da} eq 'St' ? 'stone' : $S->{Cd}{Ps} ? 'poison' :
'alive'; # 生死 Dead or Alive
    ${$V.'wn'} = $S->{wn};
    ${$V.'lz'} = $S->{ls};
    ${$V.'dr'} = $S->{dr};
    ${$V.'kl'} = $S->{kl};
    ${$V.'dt'} = $S->{dt};
    ${$V.'rw'} = $S->{rw};

    # レベル Level
    # 経験値 Experience
    # 所持金 Gold
    # 生命値 (LIF) Life
    # 生命値 (LIF) Life
    # 勝利数 Win
    # 敗北数 Lose
    # トロ一数 Draw
    # 殺害数 Kill
    # 死亡数 Death
    # 報奨金 Reward

```

```

for $Si (@{$S->{Bg}}) {
  for $i (split(/,/,$Si->{ef})) {
    ($Y,$Z) = split(/=/, $i);
    $iak = $Y eq 'lf' ? $Z : $Si->{ql} = /Rh/ ? $Y : 0;
    $idd = $Y eq 'hl' ? $Z : $Si->{ql} = /Lh|Bd/ ? $Y : 0;
    $iak = tr/a-zA-Z//d;
    $idd = tr/a-zA-Z//d;
  }
  $sts = $Si->{cl};
  ($sts .= $Si->{ql}) = tr/,//d;
  $item = $Si->{nm};
  $sts = sprintf("%04d%02d%02d%s", $Si->{bp}, $iak, $idd, $sts);
  $price = $Si->{sp};
  $new = "$item $sts";
  $new .= "$new $price" if $price;
  push(@L, $new);
}

${$V.'bg'} = join(', ', @L);

if (!$V) {
  for (@{$S->{M}}) { s/<br>$//; push(@msg, $_) }
}

}

# Sub Break Item #
sub break_item {

  $Bidice = int(rand($Bidice));

  &splititem('i');
  if (!$Bidice) {
    foreach (0 .. $#items) {
      ($item, $sts, $price) = split(/ /, $items[$_]);
      if ($sts = /Li/ && $Bidice) { next }
      if ($result == 1 && $sts = /Qw/ && $sts ! /le/) { $breakitem = 1 }
      if ($result == -1 && $sts = /Qs/ && $sts ! /Qw/ && $sts ! /le/)
        { $breakitem = 1 }
      elseif ($result == -1 && $sts = /Qa/ && $sts ! /le/) { $breakitem = 1 }
      if ($result == 0 && $sts = /Qt/ && $sts ! /le/) { $breakitem = 1 }
      if ($breakitem) {
        $sab -= substr($sts, 4, 2); $sdb -= substr($sts, 6, 2);
        splice(@items, $_, 1); &joinitem('i');
        $sp = s/Qb//g if $sts = /Bw/;
        $sp = s/Qk//g if $sts = /Kt/;
        $sp = s/Qx//g if $sts = /Ax/;
        $sp = s/Qm//g if $sts = /Sw/;
        push (@msg, "$item が壊れました "); last;
      }
    }
  }
}

}

1;

```

・ おかしな箇所があったら指摘して下さい - (2007 年 05 月 21 日 10 時 28 分 35 秒)