```
#property indicator_chart_window

extern string _Ðó÷íîé_ââîä_ìàñøòàáà_;


extern double   pH1=0;   // Ìàñøòàá Ãàííà íà H1 (0 - àâòîìàòè÷åñêèé ðàñ÷åò)

extern double   pH4=0;   // Ìàñøòàá Ãàííà íà H4 (0 - àâòîìàòè÷åñêèé ðàñ÷åò) íà òåêóóþ
íåäåëþ

extern double   pH4W_1=0; // Ìàñøòàá Ãàííà íà H4 (0 - àâòîìàòè÷åñêèé ðàñ÷åò) íà ïðåäûäóóþ
íåäåëè

extern double   pH4W_2=0; // Ìàñøòàá Ãàííà íà H4 (0 - àâòîìàòè÷åñêèé ðàñ÷åò) íà 2 íåäåëè
íàçàä


extern string _Ìàñøòàá_íåä_óã_äëÿ_Í1_ñ_Í4_;

extern bool H4_H1 = False;   // Âêëþ÷åíèå îòîáðàæåíèÿ íåäåëüíûõ óãëîâ Í4 íà H1


extern string _Ñìåùåíèå_GMT_;

extern int     DiffTime = 0;     // Ðàçíèöà âðåìåíè òåðìèíàëà è UTC


extern string ___Âðåìÿ_îòêðûòèÿ_ñåññèé___;

extern int Àçèÿ=2;     // Íà÷àëî äíÿ

extern int Åâðîïà=7;   // Íà÷àëî Åâðîïåéñêîé ñåññèè  - îòêðûòèå Ôðàíêôóðòñêîé áèðæè (ïî UTC)

extern int Àìåðèêà=13; // Íà÷àëî Àìåðèêàíñêîé ñåññèè - îòêðûòèå Íüþ-Éîðêñêîé áèðæè (ïî UTC)


extern string __Ïîñòðîåíèå_óãëîâ__;

extern bool    NextWeek=True;      // True - ïåðåõîä â ñóááîòó íà ïîíåäåëüíèê

extern bool    NextDay=False;      // True - ïåðåõîä íà äåíü âïåðåä

extern bool    ExtDay=False;       // True - óäëèííåíèå äíåâíîé ëèíèè íà äåíü âïåðåä
```

```
extern bool     FromShadow=False;     // Åñëè Äà - îò çàñòðîéóííîâ ïîñëåäíåé ñâå÷è, Íåò - îò
òåëà.

extern bool     Line=False;           // Âêëþ÷åíå, îòêëþ÷åíå óãëîâ òåêóùåãî è â÷åðàøíåãî äíÿ

extern bool     Day_minmax=True;      // Ìèí-ìàêñ ïðîøëîãî äíÿ

extern bool     Wik_op=True;          // Óãîë òåêóùåé íåäåëè

extern bool     Wik_minmax=False;     // Ìèí-ìàêñ òåêóùåé íåäåëè

extern bool     Wik_1_op=True;        // Óãîë ïðîøëîé íåäåëè

extern bool     Wik_1_minmax=True;    // Ìèí-ìàêñ ïðîøëîé íåäåëè

extern bool     Wik_2_minmax=False;   // Óãîë 2 íåäåëè íàçàä íåäåëè

extern bool     Wik_3_minmax=False;   // Óãîë 3 íåäåëè íàçàä íåäåëè


extern string   __Ïîñòðîåíå_ñåòîê__= "Äëÿ ýêñïåðèìåíòîâ ñ ñåòêàìè îòêëþ÷èòü Grid_standard";

extern bool     Grid_standard=True;       // Ñòàíäàðòíîå ïîñòðîåíå ñåòîê

extern bool     Grid_today=False;         // Âêëþ÷åíå, îòêëþ÷åíå ñåãîäíÿøåé ñåòêè

extern bool     Grid_yesterday=False;     // Âêëþ÷åíå, îòêëþ÷åíå ñåòêè â÷åðàøíåãî äíÿ

extern bool     Grid_double_H_1=False;    // Åñëè Äà - òî ñåòêà óäâîåííàÿ

extern bool     Grid_double_H_4=False;    // Åñëè Äà - òî ñåòêà óäâîåííàÿ

extern bool     Day_before_yesterday=False; // Åñëè Äà - òî ïîçàâ÷åðà è â÷åðà, Íåò - â÷åðà è
ñåãîäíÿ.

extern bool     Last_week=False;          // Åñëè Äà - òî ñìåùåíèå íà îäíó íåäåëþ íàçàä.


extern string   ____Âûáîð_Ôèáî____ = "0-íåò,1-Ôèáî,2-Ãàíí";

extern int   FiboSelect  = 1;        // 0-íåò,1 - Ôèáî, 2 - Gann.

extern color FiboColor   = Black;    // Öâåò ëèíééêè

extern int   FiboStyle  = 1;         // Ñòèëü ëèíèè ëèíééêè

extern int   FiboWight  = 0;         // Òîëùèíà ëèíèè ëèíééêè


extern string   ____Ïîñòðîåíå_êàíàëîâ____ = "false-íåò,true-ïïîñòðîåíå";
```

```
extern bool  Opening_yesterday   = False;

extern bool  High_Low           = False;


extern string ____Êàíàëû_ââåðõ;


extern color ChColorOpenUp   = ForestGreen;  // Öâåò êàíàëà ââåðõ

extern color ChColorMax     = FireBrick;  // Öâåò êàíàëà ââåðõ

extern int   ChStyleMax     = 1;        // Ñòèëü ëèíèè êàíàëà

extern int   ChWightMax     = 1;         // Òîëùèíà ëèíèè êàíàëà


extern string ____Êàíàëû_âíèç;


extern color ChColorOpenDown   = DeepPink;  // Öâåò êàíàëà âíèç

extern color ChColorMin       = Indigo;  // Öâåò êàíàëà âíèç

extern int   ChStyleMin       = 1;      // Ñòèëü ëèíèè êàíàëà

extern int   ChWightMin       = 1;      // Òîëùèíà ëèíèè êàíàëà


extern string _Ïîñòðîåíèå_S_R_min_max_ = "false-íåò,true-ïîñòðîåíå";

extern bool  S_R   = false;

extern color R_Color  = Blue;    // Öâåò ñîïðîòèâëåíèÿ

extern color S_Color  = Red;     // Öâåò ïîääåðæêè


extern string ____Ñîïðîòèâëåíèå;


extern int   S_R_Max_Style  = 0;      // Ñòèëü ñîïðîòèâëåíèÿ

extern int   S_R_Max_Wight  = 1;      // Òîëùèíà ñîïðîòèâëåíèÿ
```

```
extern string _____Ïîäääðæêà;


extern int   S_R_Min_Style  = 0;      // Ñòèëü ïïääðæêè

extern int   S_R_Min_Wight  = 1;      // Òîëùèíà ïïääðæêè


extern string _Ïîñòðîåíèå_S_R_Fibo_ = "false-íåò,true-ïîñòðîåíå";


extern bool  S_R_Fibo       = false;

extern color Op_Day_Color    = DarkGoldenrod; // Öâåò îòêðûòèÿ äíÿ

extern color R_Fibo_Color    = RoyalBlue;    // Öâåò ñîïðîòèâëåíèÿ

extern color S_Fibo_Color    = DeepPink;     // Öâåò ïïääðæê

extern int   S_R_Fibo_Style  = 2;        // Ñòèëü

extern int   S_R_Fibo_Width  = 1;         // Òîëùèíà

extern int   S_R_Fibo_Ray    = 0;         // Ëó÷


double FiboLevel[]       = { -4.00  ,  -3.618  ,  -3.382  ,  -3.236  ,  -3.0  ,  -2.618  ,
-2.382  ,  -2.236  ,  -2.0  ,  -1.618  ,  -1.382  ,  -1.236  ,  -1.0  ,  -0.618  ,  -
0.382  ,  -0.236  ,  0    ,  0.118    ,  0.236    ,  0.382    ,  0.5  ,  0.618
,  0.764    ,  0.882    ,  1    ,  1.236  ,  1.382  ,  1.618  ,  2.0  ,  2.236  ,
2.382  ,  2.618  ,  3.0  ,  3.236  ,  3.382  ,  3.618  ,  4.0  ,  4.236  ,  4.382  ,
4.618  ,  5.0  ,  100000};

string FiboDescription[]     = {" -500 %$", " -461.8 %$", " -438.2 %$", " -423.6 %$", " -400
%$", " -361.8 %$", " -338.2 %$", " -323.6 %$", " -300 %$", " -261.8 %$", " -238.2 %$", " -
223.6 %$", " -200 %$", " -161.8 %$", " -138.2 %$", " -123.6 %$", " 0---100 %$", " 11.8---
88.2 %$", " 23.6---76.4 %$", " 38.2---61.8 %$", " 50 %$", " 61.8---38.2 %$", " 76.4---23.6
%$", " 88.2---11.8 %$", " 100---0 %$", " 123.6 %$", " 138.2 %$", " 161.8 %$", " 200 %$",
" 223.6 %$", " 238.2 %$", " 261.8 %$", " 300 %$", " 323.6 %$", " 338.2 %$", " 361.8 %$",
" 400 %$", " 423.6 %$", " 438.2 %$", " 461.8 %$", " 500 %$"};

double FiboGann[]        = { -1.75  ,  -1.666  ,  -1.5  ,  -1.333  ,  -1.25  ,  -1.125  ,  -1  ,
-0.875  ,  -0.75  ,  -0.666  ,  -0.5  ,  -0.375  ,  -0.333  ,  -0.25  ,  -0.125  ,  0  ,  0.125  ,
0.25  ,  0.333  ,  0.375  ,  0.5  ,  0.666  ,  0.75  ,  0.875  ,  1  ,  1.125  ,  1.25  ,
1.333  ,  1.5  ,  1.666  ,  1.75  , 1000000};    // Gann
```

```mql4
string FiboGannDescription[]  = {" -175 %$", "  -166 %$", "  -150 %$", "  -133 %$", "  -125 %$", "
-112.5 %$", "  -1 %$", "  -87.5 %$", "  -75 %$", "  -66.6 %$", "  -50 %$", "  -37.5 %$", "  -33.3
%$", "  -25 %$", "  -12.5 %$", "  0 %$", "  12.5 %$", "  25 %$", "  33.3 %$", "  37.5 %$", "  50 %$",
"  66.6 %$", "  75 %$", "  87.5 %$", "  1 %$", "  112.5 %$", "  125 %$", "  133 %$", "  150 %$", "
166 %$", "  175 %$"};// Gann - íàäïèñè


double DT, P;

string Pr = "G_L_";

int    TimeD0; //íà÷àëî òåêóùåãî äíÿ


//+------------------------------------------------------------+
//| Custom indicator initialization function                   |
//+------------------------------------------------------------+
int init()
  {
//----

   return(0);
  }
//+------------------------------------------------------------+
//| Custom indicator deinitialization function                 |
//+------------------------------------------------------------+
int deinit()
  {
  Delete_My_Obj(Pr);
   return(0);
  }
//+------------------------------------------------------------+
//| Custom indicator iteration function                        |
```

```
//+---------------------------------------------------------+

int start()

 {

// ñìåùåíèå GMT

  DT = DiffTime*3600;

// âðåìÿ íà÷àëà òåêóùåãî äíÿ

  if (Bars < 1440 / Period()) return (0);

  int hour = iTime(NULL, 0, 0);

  int Bar_shift_hour = iBarShift(NULL, PERIOD_D1, hour);

  TimeD0 = iTime(NULL, PERIOD_D1, Bar_shift_hour) + DT;

  if (NextWeek==True)

   { if (TimeDayOfWeek(TimeD0)==6)

     if (TimeDayOfWeek(TimeLocal())==0 || TimeDayOfWeek(TimeLocal())==1 ||
TimeDayOfWeek(TimeLocal())==6) TimeD0=TimeD0+86400*3;

    }

  if (TimeDayOfWeek(TimeD0)==0) TimeD0=TimeD0+86400;

  if (NextDay==True) TimeD0=TimeD0+86400;


//---- indicators


  Delete_My_Obj(Pr);

  if(Period()<=PERIOD_H1){WriteGH1();}

  if(Period()==PERIOD_H4){WriteGH4();}


   return(0);

 }


//+---------------------------------------------------------+
```

```c
int WriteGH1()
 {
 int TimeD00;  //íà÷àëî çàâòðàøíåãî äíÿ
 int TimeD000; //íà÷àëî ïîñëåçàâòðàøíåãî äíÿ
 int TimeD1;  //íà÷àëî ïðîøëîãî äíÿ
 int TimeD2;  //íà÷àëî ïîçàïðîøëîãî äíÿ
 int minD2;  //min áàð ïîçàïðîøëîãî äíÿ
 int maxD2;  //max áàð ïîçàïðîøëîãî äíÿ
 int minD1;  //min áàð ïðîøëîãî äíÿ
 int maxD1;  //max áàð ïðîøëîãî äíÿ
 int TimeN0;  //íà÷àëî òåêóùåé íåäåëè
 int minN0;  //min áàð òåêóùåé íåäåëè
 int maxN0;  //max áàð òåêóùåé íåäåëè
 int TimeN1;  //íà÷àëî ïðîøëîé íåäåëè
 int minN1;  //min áàð ïðîøëîé íåäåëè
 int maxN1;  //max áàð ïðîøëîé íåäåëè
 int TimeN2;  //íà÷àëî 2 íåäåëè
 int minN2;  //min áàð 2 íåäåëè
 int maxN2;  //max áàð 2 íåäåëè
 int TimeN3;  //íà÷àëî 3 íåäåëè
 int minN3;  //min áàð 3 íåäåëè
 int maxN3;  //max áàð 3 íåäåëè
 int TimeN4;  //íà÷àëî 3 íåäåëè
 int minN4;  //min áàð 3 íåäåëè
 int maxN4;  //max áàð 3 íåäåëè
 int i=0, j;
```

```
    int D;

    double P, Pr1, Pr2, Pr3, MD, MYD, MW, MW_1, MW_2;

    double rez, h, l, r, t;


// âðåìÿ íà÷àëà ïðîøëîãî äíÿ

    TimeD1=TimeD0-86400;

    if (TimeDayOfWeek(TimeD0)==0) TimeD1=TimeD0-86400*2;

    if (TimeDayOfWeek(TimeD0)==1) TimeD1=TimeD0-86400*3;


// âðåìÿ íà÷àëà ïîçàïðîøëîãî äíÿ

    TimeD2=TimeD1-86400;


// âðåìÿ íà÷àëà çàâòðàøíåãî äíÿ

    TimeD00=TimeD0+86400;


// âðåìÿ íà÷àëà çàâòðàøíåãî èëè ïîñëåçàâòðàøíåãî äíÿ

    if (ExtDay==False) TimeD000=TimeD00;

    else        TimeD000=TimeD00+86400;



// âðåìÿ íà÷àëà òåêóùåé íåäåëè

    D=TimeDayOfWeek(TimeD0);

    TimeN0=TimeD0-(D-1)*86400;


// âðåìÿ íà÷àëà 4 íåäåëè

    TimeN1=TimeN0-86400*7;
```

```
// âðåìÿ íà÷àëà 3 íåäåëè

   TimeN2=TimeN1-86400*7;


// âðåìÿ íà÷àëà 2 íåäåëè

   TimeN3=TimeN2-86400*7;


// âðåìÿ íà÷àëà 2 íåäåëè

   TimeN4=TimeN3-86400*7;


// ìèí/ìàêñ ïðîøëîãî äíÿ

   int BarD0=iBarShift(NULL,0,TimeD0+DT);

   int BarD1=iBarShift(NULL,0,TimeD1+DT);

     i=BarD1-BarD0;

     maxD1=iHighest(NULL, 0, MODE_HIGH,i, BarD0+1);

     minD1=iLowest (NULL, 0, MODE_LOW ,i, BarD0+1);

   int BarD2=iBarShift(NULL,0,TimeD2+DT);

     i=BarD2-BarD1;

     maxD2=iHighest(NULL, 0, MODE_HIGH,i, BarD1+1);

     minD2=iLowest (NULL, 0, MODE_LOW ,i, BarD1+1);


// ìèí/ìàêñ òåêóùåé íåäåëè

   int BarN0=iBarShift(NULL,0,TimeN0+DT);

   j=iBarShift(NULL,0,TimeD00+DT);

   i=BarN0-j;

     maxN0=iHighest(NULL, 0, MODE_HIGH,i, j+1);

     minN0=iLowest (NULL, 0, MODE_LOW ,i, j+1);
```

```
// ìèí/ìàêñ 1 íåäåëè

   int BarN1=iBarShift(NULL,0,TimeN1+DT);

   j=iBarShift(NULL,0,TimeN0+DT);

   i=BarN1-j;

      maxN1=iHighest(NULL, 0, MODE_HIGH,i, j+1);

      minN1=iLowest (NULL, 0, MODE_LOW ,i, j+1);


// ìèí/ìàêñ 2 íåäåëè

   int BarN2=iBarShift(NULL,0,TimeN2+DT);

   j=iBarShift(NULL,0,TimeN1+DT);

   i=BarN2-j;

      maxN2=iHighest(NULL, 0, MODE_HIGH,i, j+1);

      minN2=iLowest (NULL, 0, MODE_LOW ,i, j+1);


// ìèí/ìàêñ 3 íåäåëè

   int BarN3=iBarShift(NULL,0,TimeN3+DT);

   j=iBarShift(NULL,0,TimeN2+DT);

   i=BarN3-j;

      maxN3=iHighest(NULL, 0, MODE_HIGH,i, j+1);

      minN3=iLowest (NULL, 0, MODE_LOW ,i, j+1);


// ìèí/ìàêñ 4 íåäåëè

   int BarN4=iBarShift(NULL,0,TimeN4+DT);

   j=iBarShift(NULL,0,TimeN3+DT);

   i=BarN3-j;

      maxN4=iHighest(NULL, 0, MODE_HIGH,i, j+1);

      minN4=iLowest (NULL, 0, MODE_LOW ,i, j+1);
```

```
// íà÷àëüíûå òî÷êè íåäåëü ïî òåíÿì

if (Close[BarN1]>Open[BarN1])

  Pr2=High[BarN1];

 else Pr2=Low[BarN1];


if (Close[BarN0]>Open[BarN0])

  Pr1=High[BarN0];

 else Pr1=Low[BarN0];



if (pH1==0)

 {

  h=MathAbs(NormalizeDouble(High[maxD1],4)-NormalizeDouble(Low[minD1],4))/Point;

  r=MathPow(24,2);

  rez=MathPow(h,2);

  MD=MathSqrt(rez+r)/100*Period()/60;

 }

else MD=pH1;

if (Day_before_yesterday)

 {

  h=MathAbs(NormalizeDouble(High[maxD2],4)-NormalizeDouble(Low[minD2],4))/Point;

  r=MathPow(24,2);

  rez=MathPow(h,2);

  MYD=MathSqrt(rez+r)/100*Period()/60;
```

```
    MD=MYD;

   }


 if (pH4==0)

  {

   if (H4_H1)

    {

     h=MathAbs(NormalizeDouble(High[maxN1],4)-NormalizeDouble(Low[minN1],4))/Point;

      r=MathPow(120,2);

      rez=MathPow(h,2);

      MW=MathSqrt(rez+r)/100*Period()/240;

     }

    else

     {

      MW=MD;

     }

   }

 else  MW=pH4/4;

 if (pH4W_1==0)

    {

     h=MathAbs(NormalizeDouble(High[maxN2],4)-NormalizeDouble(Low[minN2],4))/Point;

      r=MathPow(120,2);

      rez=MathPow(h,2);

      MW_1=MathSqrt(rez+r)/100*Period()/240;

     }

 else   MW_1=pH4W_1/4;

 if (pH4W_2==0)
```

```
  {

  h=MathAbs(NormalizeDouble(High[maxN3],4)-NormalizeDouble(Low[minN3],4))/Point;

  r=MathPow(120,2);

  rez=MathPow(h,2);

  MW_2=MathSqrt(rez+r)/100*Period()/240;

  }

else   MW_2=pH4W_2/4;


 Comment (WindowExpertName(         ),"\n","\n",

      "Ìàñøòàá       = ",DoubleToStr(NormalizeDouble(MD,4),4),"\n");


// ëèíèè âðåìåíè

 VLine("Àçèÿ",TimeD0+(60*60*Àçèÿ),DodgerBlue,STYLE_DOT,1,31);

 VLine("Åâðîïà",TimeD0+(60*60*Åâðîïà),DodgerBlue,STYLE_DOT,1,31);

 VLine("Àìåðèêà",TimeD0+(60*60*Àìåðèêà),DodgerBlue,STYLE_DOT,1,31);

 VLine("Íà÷àëî òåêóùåãî äíÿ",TimeD0,Maroon,STYLE_DOT,1,31);

 VLine("Êîíåö òåêóùåãî äíÿ",TimeD00,Maroon,STYLE_DOT,1,31);

 VLine("Êîíåö ñëåäóþùåãî äíÿ",TimeD00+(60*60*24),Maroon,STYLE_DOT,1,31);

 VLine("Íà÷àëî òåêóùåé íåäåëè",TimeN0,Red,STYLE_DOT,1,31);

 VLine("Íà÷àëî ïðîøëîé íåäåëè",TimeN1,Red,STYLE_DOT,1,31);



  if (FromShadow==False)

  {

   Pr2=Close[BarN1+1];

   Pr1=Close[BarN0+1];

  }
```

```
//Óãîë òåêóùåé íåäåëè

if (Wik_op)

   GannLine ("Óãîë òåêóùåé íåäåëè","Line",TimeN0,Pr1,TimeD00,MW,Turquoise,0,1,31);


//ìèí ìàêñ òåêóùåé íåäåëè

if (Wik_minmax)

{

   GannLine ("Ìàêñ òåêóùåé
íåäåëè","Line",Time[maxN0],High[maxN0],TimeD00,MW,DeepPink, 1,1,31);

   GannLine ("Ìèí òåêóùåé íåäåëè","Line",Time[minN0],Low[minN0], TimeD00,MW,DeepPink,
1,1,31);

}


//óãîë ïðîøëîé íåäåëè

if (Wik_1_op)

   GannLine ("Óãîë ïðîøëîé íåäåëè","Line",TimeN1,     Pr2,TimeD00,MW,Black,0,1,31);


//ìèí ìàêñ ïðîøëîé íåäåëè

if (Wik_1_minmax)

{

   GannLine ("Ìàêñ ïðîøëîé íåäåëè","Line",Time[maxN1],High[maxN1],TimeD00,MW,Gold,
1,1,31);

   GannLine ("Ìèí ïðîøëîé íåäåëè","Line",Time[minN1],Low[minN1], TimeD00,MW,Gold, 1,1,31);

}


//ìèí ìàêñ 2 íåäåëè

if (Wik_2_minmax)
```

```
   {

     GannLine ("Ìàêñ 2 íåäåëè","Line",Time[maxN2],High[maxN2],TimeD00,MW_1,White,
1,1,31);

     GannLine ("Ìèí 2 íåäåëè","Line",Time[minN2],Low[minN2], TimeD00,MW_1,White, 1,1,31);

   }


   //ìèí ìàêñ 3 íåäåëè

   if (Wik_3_minmax)

   {

     GannLine ("Ìàêñ 3 íåäåëè","Line",Time[maxN3],High[maxN3],TimeD00,MW_2,SaddleBrown,
1,1,31);

     GannLine ("Ìèí 3 íåäåëè","Line",Time[minN3],Low[minN3], TimeD00,MW_2,SaddleBrown,
1,1,31);

   }


   //â÷åðàøíèé óãîë

   if (FromShadow==TRUE)

   {

       if (Close[BarD2]>Open[BarD2])

         Pr3=High[BarD2];

         else Pr3=Low[BarD2];


       if (Close[BarD1]>Open[BarD1])

         Pr2=High[BarD1];

         else Pr2=Low[BarD1];


       if (Close[BarD0]>Open[BarD0])

         Pr1=High[BarD0];

         else Pr1=Low[BarD0];
```

```
  }
 else

   {

    Pr3=Close[BarD2+1];

    Pr2=Close[BarD1+1];

    Pr1=Close[BarD0+1];

   }
//ñòàíäàðòíîå ïîñòðîåíèå ñåòîê
 if (Grid_standard)
 {
  Grid_today=False;

  Grid_yesterday=False;

  Grid_double_H_1=False;

  Day_before_yesterday=False;

  GannLine ("Ñåòêà ïî óãëó â÷åðàøíåãî äíÿ","Grid",TimeD1,Pr2,TimeD00,MD,Maroon,0,1,31);

  GannLine ("Ñåòêà ïî óãëó òåêóùåãî äíÿ","Grid",TimeD0,Pr1,TimeD00,MD,Navy, 0,1,31);

 }


//óãîë â÷åðàøíåãî äíÿ
 if (Line)GannLine ("óãîë â÷åðàøíåãî äíÿ","Line",TimeD1,  Pr2,   TimeD00,MD,Maroon,0,1,31);


//óãîë òåêóùåãî äíÿ
 if (Line)GannLine ("Óãîë òåêóùåãî äíÿ","Line",TimeD0,Pr1,     TimeD000,MD,Navy, 0,1,31);


 if (Day_before_yesterday) //â÷åðàøíèé óãîë è ïîçàâ÷åðàøíèé óãîë
 {
  if (Grid_double_H_1)
```

```
   {

     if (Grid_yesterday)GannLine ("Ñåòêà ïî óãëó ïîçàâ÷åðàøíåãî
äíÿ","Grid",TimeD2,Pr3,TimeD0,MD,Maroon,0,2,31);

     if (Grid_today)GannLine ("Ñåòêà ïî óãëó â÷åðàøíåãî
äíÿ","Grid",TimeD1,Pr1,TimeD00,MD,Navy, 0,2,31);

   }

   else

   {

     if (Grid_yesterday)GannLine ("Ñåòêà ïî óãëó ïîçàâ÷åðàøíåãî äíÿ","Grid",TimeD2,  Pr3,
TimeD1,MD,Maroon,0,1,31);

     if (Grid_today)GannLine ("Ñåòêà ïî óãëó â÷åðàøíåãî äíÿ","Grid",TimeD1,  Pr2,
TimeD0,MD,Navy, 0,1,31);

   }

 }

   else //â÷åðàøíèé óãîë è ñåãîäíÿøíèé

   {

   if (Grid_double_H_1)

   {

     if (Grid_yesterday)GannLine ("Ñåòêà ïî óãëó â÷åðàøíåãî
äíÿ","Grid",TimeD1,Pr2,TimeD00,MD,Maroon,0,2,31);

     if (Grid_today)GannLine ("Ñåòêà ïî óãëó òåêóùåãî
äíÿ","Grid",TimeD0,Pr1,TimeD00+86400,MD,Navy, 0,2,31);

   }

   else

   {

     if (Grid_yesterday)GannLine ("Ñåòêà ïî óãëó â÷åðàøíåãî
äíÿ","Grid",TimeD1,Pr2,TimeD0,MD,Maroon,0,1,31);

     if (Grid_today)GannLine ("Ñåòêà ïî óãëó òåêóùåãî
äíÿ","Grid",TimeD0,Pr1,TimeD00,MD,Navy, 0,1,31);

   }

 }
```

```
//ìèí ìàêñ â÷åðàøíåãî äíÿ

  if (Day_minmax)

  {

  if (Day_before_yesterday)

   {

    GannLine ("Ìàêñ ïîçàâ÷åðàøíåãî äíÿ","Line",Time[maxD2],High[maxD2],TimeD0,MD,Lime,
1,1,31);

    GannLine ("Ìèí ïîçàâ÷åðàøíåãî äíÿ","Line",Time[minD2], Low[minD2],TimeD0,MD,Lime,
1,1,31);

    }

   else

    {

    GannLine ("Ìàêñ â÷åðàøíåãî äíÿ","Line",Time[maxD1],High[maxD1],TimeD00,MD,Lime,
1,1,31);

    GannLine ("Ìèí â÷åðàøíåãî äíÿ","Line",Time[minD1], Low[minD1],TimeD00,MD,Lime,
1,1,31);

    }

   }


  //ôèáîíàíëû

  if (High_Low)

  {

  Channel ("Êàíàë ïî ìèí_ìàêñó â÷åðàøíåãî äíÿ+", Time[maxD1],High[maxD1], Time[minD1],
Low[minD1],  MD,  31, ChColorMax, ChStyleMax, ChWightMax );

  Channel ("Êàíàë ïî ìèí_ìàêñó â÷åðàøíåãî äíÿ-", Time[maxD1],High[maxD1], Time[minD1],
Low[minD1],  -MD,  31, ChColorMin, ChStyleMin, ChWightMin  );

  }

  if (Opening_yesterday)

  {
```

```
    Channel ("Êàíàë ïî îòêðûòèþ â÷åðàøíåãî äíÿ+", TimeD1,  Pr2, TimeD0+86400*3, Pr2,  MD,  31,
ChColorOpenUp, ChStyleMax, ChWightMax );

    Channel ("Êàíàë ïî îòêðûòèþ â÷åðàøíåãî äíÿ-", TimeD1,  Pr2, TimeD0+86400*3, Pr2,  -MD,
31, ChColorOpenDown, ChStyleMin, ChWightMin  );

    }


 //ñîïðîòèâëåíèÿ è ïïääåðæêè

  if (S_R)

  {

    ÍLine ("Ìàêñ ïðîøëîé íåäåëè", High[maxN1], S_R_Max_Style, S_R_Max_Wight,31);

    ÍLine ("Ìèí ïðîøëîé íåäåëè", Low[minN1], S_R_Min_Style, S_R_Min_Wight,31);


    ÍLine ("Ìàêñ âòîðîé íåäåëè", High[maxN2], S_R_Max_Style, S_R_Max_Wight,31);

    ÍLine ("Ìèí âòîðîé íåäåëè", Low[minN2], S_R_Min_Style, S_R_Min_Wight,31);


    ÍLine ("Ìàêñ òðåòüåé íåäåëè", High[maxN3], S_R_Max_Style, S_R_Max_Wight,31);

    ÍLine ("Ìèí òðåòüåé íåäåëè", Low[minN3], S_R_Min_Style, S_R_Min_Wight,31);


    ÍLine ("Ìàêñ ÷åòâ¸ðòîé íåäåëè", High[maxN4], S_R_Max_Style, S_R_Max_Wight,31);

    ÍLine ("Ìèí ÷åòâ¸ðòîé íåäåëè", Low[minN4], S_R_Min_Style, S_R_Min_Wight,31);

    }


 // Ïîñòðîåíèå äíåâíûõ ôèáî_S_R

  if (S_R_Fibo)

  {

    P = Open[BarD0];

    Fibo_S_R("Îòêðûòèå äíÿ", P, TimeD0, P, TimeD00, 31);

    Fibo_S_R("R_1", P, TimeD0, P+13*Point*10, TimeD00, 31);
```

```
        Fibo_S_R("S_1", P, TimeD0, P-13*Point*10, TimeD00, 31);

        Fibo_S_R("R_2", P, TimeD0, P+21*Point*10, TimeD00, 31);

        Fibo_S_R("S_2", P, TimeD0, P-21*Point*10, TimeD00, 31);

        Fibo_S_R("R_3", P, TimeD0, P+34*Point*10, TimeD00, 31);

        Fibo_S_R("S_3", P, TimeD0, P-34*Point*10, TimeD00, 31);

        Fibo_S_R("R_4", P, TimeD0, P+55*Point*10, TimeD00, 31);

        Fibo_S_R("S_4", P, TimeD0, P-55*Point*10, TimeD00, 31);

        Fibo_S_R("R_5", P, TimeD0, P+89*Point*10, TimeD00, 31);

        Fibo_S_R("S_5", P, TimeD0, P-89*Point*10, TimeD00, 31);

        Fibo_S_R("R_6", P, TimeD0, P+144*Point*10, TimeD00, 31);

        Fibo_S_R("S_6", P, TimeD0, P-144*Point*10, TimeD00, 31);

   }


 //ôèáî óðîâíè îò ìèí ìàêñ â÷åðàøíåãî äíÿ

 if (FiboSelect == 1)

 {

Fibo("Fibo",FiboLevel,FiboDescription,Time[maxD1],High[maxD1],Time[minD1],Low[minD1],FiboColor,FiboStyle,FiboWight,31);

 }

 if (FiboSelect == 2)

 {

Fibo("Gann",FiboGann,FiboGannDescription,Time[maxD1],High[maxD1],Time[minD1],Low[minD1],FiboColor,FiboStyle,FiboWight,31);

 }


  WindowRedraw();
```

```c
return(0);
 }


 //---------------------------------------------------------------------------
 int WriteGH4()
  {
  int TimeD00;//íà÷àëî áóäóùåãî äíÿ
  int TimeN00;//íà÷àëî áóäóùåé íåäåëè
  int TimeN0; //íà÷àëî òåêóùåé íåäåëè
  int TimeN1; //íà÷àëî ïðîøëîé íåäåëè
  int TimeN2; //íà÷àëî ïðîøëîé íåäåëè
  int TimeM; //íà÷àëî ìåñÿöà
  int minN1; //min áàð ïðîøëîé íåäåëè
  int maxN1; //max áàð ïðîøëîé íåäåëè
  int minN2; //min áàð ïîçàïðîøëîé íåäåëè
  int maxN2; //max áàð ïîçàïðîøëîé íåäåëè
  int i=0, j;
  int D;
  double Pr1, Pr2, Pr3, MW, MV;
  double rez, h, l, r, t;


// âðåìÿ íà÷àëà çàâòðàøíåãî äíÿ
    if (ExtDay==False) TimeD00=TimeD0+86400;
    else          TimeD00=TimeD0+86400*2;


// âðåìÿ íà÷àëà òåêóùåé íåäåëè
  D=TimeDayOfWeek(TimeD0);
```

```
    TimeN0=TimeD0-(D-1)*86400;


// âðåìÿ íà÷àëà ïðåäûäóùåé íåäåëè

    TimeN1=TimeN0-86400*7;


// âðåìÿ íà÷àëà ïðåäûäóùåé íåäåëè

    TimeN2=TimeN1-86400*7;


//íà÷àëî áóäóùåé íåäåëè

     TimeN00=TimeN0+86400*5;


// ìèí/ìàêñ ïðîøëîé íåäåëè

    int BarN0=iBarShift(NULL,0,TimeN0+DT);

    int BarN1=iBarShift(NULL,0,TimeN1+DT);

    j=iBarShift(NULL,0,TimeN0+DT);

    i=BarN1-j;

        maxN1=iHighest(NULL, 0, MODE_HIGH,i, j+1);

        minN1=iLowest (NULL, 0, MODE_LOW ,i, j+1);

    int BarN2=iBarShift(NULL,0,TimeN2+DT);

    j=iBarShift(NULL,0,TimeN1+DT);

    i=BarN2-j;

        maxN2=iHighest(NULL, 0, MODE_HIGH,i, j+1);

        minN2=iLowest (NULL, 0, MODE_LOW ,i, j+1);


// âû÷èñëåíèå ïàðàìåòðà

        if (Close[BarN1]>Open[BarN1])

          Pr2=High[BarN1];
```

```
        else Pr2=Low[BarN1];


    if (Close[BarN0]>Open[BarN0])

      Pr1=High[BarN0];

      else Pr1=Low[BarN0];


if (pH4==0)

 {

  h=MathAbs(NormalizeDouble(High[maxN1],4)-NormalizeDouble(Low[minN1],4))/Point;

  r=MathPow(120,2);

  rez=MathPow(h,2);

  MW=MathSqrt(rez+r)/100*Period()/240;

 }

else MW=pH4;

if (Last_week)

 {

  h=MathAbs(NormalizeDouble(High[maxN2],4)-NormalizeDouble(Low[minN2],4))/Point;

  r=MathPow(120,2);

  rez=MathPow(h,2);

  MV=MathSqrt(rez+r)/100*Period()/240;

  MW=MV;

 }



Comment (WindowExpertName(        ),"\n","\n",

      "Ìàñøòàá ïðîøëîé íåäåëè = ",DoubleToStr(NormalizeDouble(MW,4),4),"\n");
```

```
// ëèíèè âðåìåíè

VLine("Îêîí÷àíèå òåêóùåé íåäåëè",TimeN00,DodgerBlue,STYLE_DOT,1,OBJ_PERIOD_H4);

VLine("Íà÷àëî òåêóùåé íåäåëè",TimeN0,DodgerBlue,STYLE_DOT,1,OBJ_PERIOD_H4);

VLine("Íà÷àëî ïðîøëîé íåäåëè",TimeN1,DodgerBlue,STYLE_DOT,1,OBJ_PERIOD_H4);

VLine("Íà÷àëî òåêóùåãî äíÿ",TimeD0,PaleGreen,STYLE_DOT,1,OBJ_PERIOD_H4);

VLine("Îêîí÷àíèå òåêóùåãî äíÿ",TimeD00,PaleGreen,STYLE_DOT,1,OBJ_PERIOD_H4);


// óãîë ïðîøëîé íåäåëè
if (FromShadow==False)
  {
   Pr1=Close[BarN0+1];
   Pr2=Close[BarN1+1];
   Pr3=Close[BarN2+1];
  }


 if (Last_week)
  {
   if (Grid_double_H_4)
    {
     GannLine ("Óãîë ïîçàïðîøëîé íåäåëè","Grid",TimeN2,
Pr3,TimeN1+86400*5,MW,Maroon,0,2,OBJ_PERIOD_H4);

     GannLine ("Óãîë ïðîøëîé íåäåëè","Grid",TimeN1,
Pr2,TimeN0,MW,Navy,0,1,OBJ_PERIOD_H4);

    }
   else
    {
     GannLine ("Óãîë ïîçàïðîøëîé íåäåëè","Grid",TimeN2,
Pr3,TimeN1,MW,Maroon,0,1,OBJ_PERIOD_H4);
```

```
      GannLine ("Óãîë ïðîøëîé íåäåëè","Grid",TimeN1,
Pr2,TimeN0,MW,Navy,0,1,OBJ_PERIOD_H4);

   }

  }

  else

  {

  if (Grid_double_H_4)

   {

    GannLine ("Óãîë ïðîøëîé íåäåëè","Grid",TimeN1,
Pr2,TimeN0+86400*5,MW,Maroon,0,2,OBJ_PERIOD_H4);

    GannLine ("Óãîë òåêóùåé íåäåëè","Grid",TimeN0,
Pr1,TimeN00,MW,Navy,0,1,OBJ_PERIOD_H4);

   }

   else

   {

    GannLine ("Óãîë ïðîøëîé íåäåëè","Grid",TimeN1,
Pr2,TimeN0,MW,Maroon,0,1,OBJ_PERIOD_H4);

    GannLine ("Óãîë òåêóùåé íåäåëè","Grid",TimeN0,
Pr1,TimeN00,MW,Navy,0,1,OBJ_PERIOD_H4);

   }

  }


 //ìèí ìàêñ ïðîøëîé íåäåëè

   if (Last_week)

   {

    GannLine ("Max ïîçàïðîøëîé íåäåëè","Line",Time[maxN2],
High[maxN2],TimeN0,MW,Gold,0,1,OBJ_PERIOD_H4);

    GannLine ("Min ïîçàïðîøëîé íåäåëè","Line",Time[minN2],
Low[minN2],TimeN0,MW,Gold,0,1,OBJ_PERIOD_H4);

   }
```

```
    else

      {

      GannLine ("Max ïðîøëîé íåäåëè","Line",Time[maxN1],
High[maxN1],TimeN00,MW,Gold,0,1,OBJ_PERIOD_H4);

      GannLine ("Min ïðîøëîé íåäåëè","Line",Time[minN1],
Low[minN1],TimeN00,MW,Gold,0,1,OBJ_PERIOD_H4);

      }


  //ôèáîêàíàëû

    if (High_Low)

    {

    Channel ("Êàíàë ïî ìèí_ìàêñó â÷åðàøíåãî äíÿ+", Time[maxN1],High[maxN1], Time[minN1],
Low[minN1],  MW,  OBJ_PERIOD_H4, ChColorMax, ChStyleMax, ChWightMax );

    Channel ("Êàíàë ïî ìèí_ìàêñó â÷åðàøíåãî äíÿ-", Time[maxN1],High[maxN1], Time[minN1],
Low[minN1],  -MW,  OBJ_PERIOD_H4, ChColorMin, ChStyleMin, ChWightMin  );

    }

    if (Opening_yesterday)

    {

    Channel ("Êàíàë ïî îòêðûòèþ â÷åðàøíåãî äíÿ+", TimeN1,  Pr2, TimeN0+86400*7, Pr2,  MW,
OBJ_PERIOD_H4, ChColorOpenUp, ChStyleMax, ChWightMax );

    Channel ("Êàíàë ïî îòêðûòèþ â÷åðàøíåãî äíÿ-", TimeN1,  Pr2, TimeN0+86400*7, Pr2,  -MW,
OBJ_PERIOD_H4, ChColorOpenDown, ChStyleMin, ChWightMin  );

    }


  //ôèáî óðîâíè îò ìèí ìàêñ ïðîøëîé íåäåëè

  if (FiboSelect == 1)

  {


Fibo("Fibo",FiboLevel,FiboDescription,Time[maxN1],High[maxN1],Time[minN1],Low[minN1],Fi
boColor,FiboStyle,FiboWight,OBJ_PERIOD_H4);
```

```
    }

   if (FiboSelect == 2)

   {

Fibo("Gann",FiboGann,FiboGannDescription,Time[maxN1],High[maxN1],Time[minN1],Low[min
N1],FiboColor,FiboStyle,FiboWight,OBJ_PERIOD_H4);

   }


   WindowRedraw();


return(0);

  }


// ----------------------------------------------------------------------

//  Ïîñòðîåíèå ïàðû ëèíèé \ ñåòîê Ãàííà                          |

//----------------------------------------------------------------------

void GannLine (string name, string type, datetime x1, double y1, datetime x2, double Mashtab,
color Color, int Style, int Widht, int period)

  {

    if (type=="Line") ObjectCreate(Pr+name+"+",OBJ_GANNLINE,0,x1,y1,x2,0);

    if (type=="Grid") ObjectCreate(Pr+name+"+",OBJ_GANNGRID,0,x1,y1,x2,0);

     ObjectSet (Pr+name+"+",OBJPROP_COLOR,Color);

     ObjectSet (Pr+name+"+",OBJPROP_STYLE,Style);

     ObjectSet (Pr+name+"+",OBJPROP_WIDTH,Widht);

     ObjectSet (Pr+name+"+",OBJPROP_TIMEFRAMES,period);

     ObjectSet (Pr+name+"+",OBJPROP_SCALE,Mashtab);


    if (type=="Line") ObjectCreate(Pr+name+"-",OBJ_GANNLINE,0,x1,y1,x2,0);

    if (type=="Grid") ObjectCreate(Pr+name+"-",OBJ_GANNGRID,0,x1,y1,x2,0);
```

```
        ObjectSet (Pr+name+"-",OBJPROP_COLOR,Color);

        ObjectSet (Pr+name+"-",OBJPROP_STYLE,Style);

        ObjectSet (Pr+name+"-",OBJPROP_WIDTH,Widht);

        ObjectSet (Pr+name+"-",OBJPROP_TIMEFRAMES,period);

        ObjectSet (Pr+name+"-",OBJPROP_SCALE,-Mashtab);


  return(0);

  }



// ------------------------------------------------------------------------

//  Ïîñòðîåíèå ôèáîêàíàëîâ                                  |

//------------------------------------------------------------------------

void Channel (string name, datetime x1, double y1, datetime x2, double y2, double Mashtab,
int period, color ChColor, int ChStyle, int ChWidth )

{

  int BarN0=iBarShift(NULL,0,x1);

  double y3=y1 - ((double)BarN0)*Mashtab*Point;



  ObjectCreate(Pr+name,OBJ_FIBOCHANNEL,0,x1,y1, iTime( NULL, 0, 0) , y3,x2,y2 );



  ObjectSet (Pr+name,OBJPROP_LEVELCOLOR,ChColor);

  ObjectSet (Pr+name,OBJPROP_LEVELSTYLE,ChStyle);

  ObjectSet (Pr+name,OBJPROP_LEVELWIDTH,ChWidth);

  ObjectSet (Pr+name,OBJPROP_TIMEFRAMES,period);



  ObjectSet (Pr+name,OBJPROP_FIBOLEVELS,20);

  ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL +0,0);
```

```
   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+1,0.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+2,1);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+3,1.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+4,2);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+5,2.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+6,3);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+7,3.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+8,4);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+9,4.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+10,5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+11,-0.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+12,-1);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+13,-1.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+14,-2);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+15,-2.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+16,-3);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+17,-3.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+18,-4);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+19,-4.5);

   ObjectSet (Pr+name,OBJPROP_FIRSTLEVEL+20,-5);


   return(0);
}


// ------------------------------------------------------------------------
//  Ïîñòðîåíèå âåðòèêàëüíîé ëèíèè                                       |
//------------------------------------------------------------------------
```

```
void VLine (string name, datetime x1, color Color, int Style,int Back, int period)

{

  ObjectCreate(Pr+name,OBJ_VLINE,0,x1,0);

  ObjectSet(Pr+name,OBJPROP_COLOR,Color);

  ObjectSet(Pr+name,OBJPROP_STYLE,Style);

  ObjectSet(Pr+name,OBJPROP_BACK,Back);

  ObjectSet(Pr+name,OBJPROP_TIMEFRAMES,period);


  return(0);

}



// ------------------------------------------------------------------------

//  Ïîñòðîåíèå ãîðèçîíòàëüíîé ëèíèè                          |

//-------------------------------------------------------------------------

void ÍLine (string name, double y1, int Style, int Width, int period)

{

  color Color;

  if (y1 > Bid) Color = R_Color;

  if (y1 < Bid) Color = S_Color;

  ObjectCreate(Pr+name,OBJ_HLINE,0,0,y1);

  ObjectSet(Pr+name,OBJPROP_COLOR,Color);

  ObjectSet(Pr+name,OBJPROP_STYLE,Style);

  ObjectSet(Pr+name,OBJPROP_WIDTH,Width);

  ObjectSet(Pr+name,OBJPROP_TIMEFRAMES,period);


  return(0);

}
```

```
//----------------------------------------------------

// Âûâîä ôèáî óðîâíåé

//----------------------------------------------------

void Fibo(string name,  double fi_[], string fitxt_[], datetime x1, double y1, datetime x2,double
y2,color Color, int Style, int Widht, int period)

{

    int j = 0;

    int m = 0;

    int p = 0;


    while (fi_[j] < 100 )

    {

        if (fi_[j] >=0 )

        {

          p++;

        }

        else

        {

          m++;

        }

        j++;

    }

    string FiboNameP = Pr+name + "p";

    string FiboNameM = Pr+name + "m";


    ObjectCreate(FiboNameP,OBJ_FIBO,0,x1,y1,x2,y2);

    ObjectSet(FiboNameP,OBJPROP_FIBOLEVELS,p);
```

```
ObjectCreate(FiboNameM,OBJ_FIBO,0,x1,y1,x2,y2);

ObjectSet(FiboNameM,OBJPROP_FIBOLEVELS,m);

m = 0;

p = 0;

for ( int i = 0; i < j; i++ )

{


    if (fi_[i] >=0 )

    {

      ObjectSet(FiboNameP,OBJPROP_FIRSTLEVEL+p,fi_[i]);

      ObjectSetFiboDescription(FiboNameP, p, fitxt_[i]);

      p++;

    }

    else

    {

      ObjectSet(FiboNameM,OBJPROP_FIRSTLEVEL+m,fi_[i]);

      ObjectSetFiboDescription(FiboNameM, m, fitxt_[i]);

      m++;

    }


}

ObjectSet(FiboNameP,OBJPROP_LEVELCOLOR,Color);

ObjectSet(FiboNameP,OBJPROP_LEVELSTYLE,Style);

ObjectSet(FiboNameP,OBJPROP_LEVELWIDTH,Widht);

ObjectSet(FiboNameP,OBJPROP_TIMEFRAMES,period);
```

```
    ObjectSet(FiboNameM,OBJPROP_LEVELCOLOR,Color);

    ObjectSet(FiboNameM,OBJPROP_LEVELSTYLE,Style);

    ObjectSet(FiboNameM,OBJPROP_LEVELWIDTH,Widht);

    ObjectSet(FiboNameM,OBJPROP_TIMEFRAMES,period);


  return(0);

  }



//+--------------------------------------------------------------+
// Ïîñòðîåíèå äåâåíûõ ôèáî_S_R                          |
//+--------------------------------------------------------------+
void Fibo_S_R (string name, double y1, datetime x1, double y2, datetime x2, int period)
{
    color Color;
    if (y1 == y2) Color = Op_Day_Color;
    if (y1 < y2)  Color = R_Fibo_Color;
    if (y1 > y2)  Color = S_Fibo_Color;
    ObjectCreate(Pr+name,OBJ_TREND,0,x1,y2,x2,y2);
    ObjectSet(Pr+name,OBJPROP_COLOR,Color);
    ObjectSet(Pr+name,OBJPROP_STYLE,S_R_Fibo_Style);
    ObjectSet(Pr+name,OBJPROP_WIDTH,S_R_Fibo_Width);
    ObjectSet(Pr+name,OBJPROP_RAY,S_R_Fibo_Ray);
    ObjectSet(Pr+name,OBJPROP_TIMEFRAMES,period);


  return(0);

}
```

```
//+------------------------------------------------------------+

// Äåèíñòàëÿòîð îáúåêòîâ                        |

//+------------------------------------------------------------+

void Delete_My_Obj(string Prefix)

{

  for(int k = ObjectsTotal()-1; k >= 0; k --)  // Ïî êîëè÷åñòâó âñåõ îáúåêòîâ

  {

    string Obj_Name = ObjectName(k);   // Çàïðàøèâàåì èìÿ îáúåêòà

    string Head = StringSubstr(Obj_Name, 0, StringLen(Prefix));// Èçâëåêàåì ïåðâûå ñèì


    if (Head == Prefix)// Íàéäåí îáúåêò, ..

    {

      ObjectDelete(Obj_Name);

    }


  }

}

//+------------------------------------------------------------+
```